

UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO

CARRERA:
INGENIERÍA DE SISTEMAS

Trabajo de titulación previo a la obtención del título de:
INGENIERO DE SISTEMAS

TEMA:
**“ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE UN GEOPORTAL
ORIENTADO A DISPOSITIVOS MÓVILES EN SISTEMA OPERATIVO
ANDROID PARA VISUALIZAR LAS OBRAS SALESIANAS EN EL SECTOR
EDUCATIVO, PASTORAL Y AUTO GESTIONADO”.**

AUTOR:
HUGO FABIÁN CAJAMARCA QUISHPE

DIRECTOR:
ALONSO RENÉ ARÉVALO CAMPOS

Quito, mayo de 2015

DECLARATORIA DE RESPONSABILIDAD Y AUTORIZACIÓN DE USO DEL TRABAJO DE TITULACIÓN

Yo, autorizo a la Universidad Politécnica Salesiana la publicación total o parcial de este trabajo de titulación y su reproducción sin fines de lucro.

Además, declaro que los conceptos, análisis desarrollados y las conclusiones del presente trabajo son de exclusiva responsabilidad mía.

Quito, mayo, 2015

Hugo Fabián Cajamarca Quishpe
C.C. 050243842-7

DEDICATORIA

Dedico el resultado del presente trabajo a mis padres Eugenio Cajamarca y María Elena Quishpe Chasi por su amor y apoyo incondicional en todo momento en todos los ámbitos de mi vida, los mismos que nunca perdieron la fe les dedico este sueño realizado.

Hugo Fabián Cajamarca Quishpe

AGRADECIMIENTO

A la Universidad Politécnica Salesiana y a sus docentes por impartir sus conocimientos y formarme como un profesional de bien, por brindarme la oportunidad de superarme día a día.

Hugo Fabián Cajamarca Quishpe

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1	3
JUSTIFICACIÓN Y FORMULACIÓN DEL PROBLEMA.....	3
1.1 Justificación.....	3
1.2 Formulación del problema.....	3
1.3 Importancia.....	4
1.4 Objetivos	4
1.4.1 Objetivo general.	4
1.4.2 Objetivos específicos.....	4
1.5 Metodología.....	4
1.5.1 XtremeProgramming (XP)	5
1.5.2 Metodología de Programación Extrema o XP.....	5
1.5.2.1 Planeación	6
1.5.2.2 Diseño	7
1.5.2.3 Codificación	8
1.5.2.4 Pruebas	8
1.6 Alcance.....	9
1.6.1 Módulos servidor.....	9
CAPÍTULO 2	10
MARCO TEÓRICO	10
2.1 Google Android.....	10
2.1.1 Versiones Android.....	10
2.2 Códigos QR (quick response code, «código de respuesta rápida»).	11
2.2.1 Para qué sirve un código QR.....	12
2.2.2 Qué puede contener un código QR.....	12
2.3 ZXing.....	13
2.4 Apache Tomcat.....	13
2.4.1 Historia	14
2.4.2 Estructura.....	14
2.4.3 Componentes	15

2.4.4	Características y evolución.....	16
2.5	PostgreSQL 9	18
2.6	PostGIS 1.5.....	18
2.7	Quantum GIS (QGIS) 1.8.....	20
2.8	Eclipse 4.2	20
2.9	Google Maps Android API v2.....	20
2.10	Phpqrcode	21
2.11	Bootstrap	21
2.11.1	Accesibilidad	21
2.11.2	La licencia de Bootstrap	22
2.11.2.1	Lo que te permite la licencia	22
2.11.2.2	Lo que te prohíbe la licencia	22
2.12	Apache-JMeter-2.11	22
2.13	Google Maps	23
2.13.1	Desarrollo	23
2.13.2	Básicas.....	24
2.14	Google Play Services.....	24
2.15	Las pruebas y el desarrollo de software	25
CAPÍTULO 3		27
DISEÑO		27
3.1	Especificación de requisitos del Software.....	27
3.2	Diagrama de componentes	29
3.3	Diagramas de casos de uso.....	31
3.4	Diseño del sistema.....	37
3.4.1	Diseño de la base de datos.....	37
CAPÍTULO 4		40
DESARROLLO.....		40
4.1	Desarrollo del módulo de administración	40
4.1.1	Funcionalidades del módulo administrador	40
4.1.2	Acceso al sistema	40
4.2	Desarrollo módulo cliente	48

4.2.1 Clase MainActivity.....	49
4.2.2 Clase Mapa.....	49
4.2.3 Clase Mapa.....	51
4.2.4 Visualizador de obras Salesianas	52
CAPÍTULO 5	62
IMPLEMENTACIÓN Y PRUEBAS.....	62
5.1 Implementación	62
5.2 Pruebas	63
5.2.1 Introducción.....	63
5.2.2 Tipos de pruebas realizadas.....	63
5.2.3 Herramientas utilizadas	71
CONCLUSIONES.....	72
RECOMENDACIONES.....	73
LISTA DE REFERENCIAS	74
ANEXOS.....	76

ÍNDICE DE TABLAS

Tabla 1. <i>Versiones de Android</i>	11
Tabla 2. <i>Roles de los usuarios en el sistema</i>	27
Tabla 3. <i>Tarjeta CRC de la clase usuario</i>	28
Tabla 4. <i>Tarjeta CRC de la clase perfil</i>	29
Tabla 5. <i>Casos de uso roles y tareas</i>	31
Tabla 6. <i>Casos de uso administrador usuario y perfiles</i>	32
Tabla 7. <i>Especificaciones técnicas del Servidor</i>	62
Tabla 8 <i>Especificaciones técnicas del dispositivo móvil</i>	62
Tabla 9. <i>Características del equipo usado en las pruebas</i>	63
Tabla 10. <i>Prueba ingreso al sistema de administración web</i>	70
Tabla 11. <i>Pruebas de ingreso al sistema Android</i>	71

ÍNDICE DE FIGURAS

<i>Figura 1.</i> Etapas de XP	6
<i>Figura 2.</i> Código QR	12
<i>Figura 3.</i> Diagrama de componentes	30
<i>Figura 4.</i> Caso de uso ingreso al sistema.....	33
<i>Figura 5.</i> Caso de Uso administrador usuarios y perfiles	34
<i>Figura 6.</i> Caso de uso administrado usuarios y perfiles	35
<i>Figura 7.</i> Caso de uso cliente Android	36
<i>Figura 8.</i> Diseño conceptual	37
<i>Figura 9.</i> Diseño físico de la base de datos	38
<i>Figura 10.</i> Base de datos del CIMA	39
<i>Figura 11.</i> Clase autenticación	41
<i>Figura 13.</i> Clase Función BD	43
<i>Figura 14.</i> Clase función BD	44
<i>Figura 15.</i> Clase administración usuarios.....	45
<i>Figura 16.</i> Clase cabecera.....	46
<i>Figura 17.</i> Página principal perfil administrador.....	47
<i>Figura 18.</i> Página principal perfil invitado.....	48
<i>Figura 19.</i> Clase MainActivity	49
<i>Figura 20.</i> Clase Mapa.....	50
<i>Figura 21.</i> Clase Mapa.....	51
<i>Figura 22.</i> Visualizador de obras Salesianas	52
<i>Figura 23.</i> Menús y submenús.....	53
<i>Figura 24.</i> Menú administración	54
<i>Figura 25.</i> Menú administración, submenú obras Salesianas.....	55
<i>Figura 26.</i> Menú administración, submenú obras Salesianas.....	55
<i>Figura 27.</i> Pantalla inicio del cliente Android.....	56
<i>Figura 28.</i> Menú escanear código QR	56
<i>Figura 29.</i> Información contenida en el código QR	57
<i>Figura 30.</i> Opciones adicionales provistas por la librería de Android	58
<i>Figura 31.</i> Menú Mapa funcionalidad de la librería	59
<i>Figura 32.</i> Opciones de viaje provistas por la librería de Android.....	60

<i>Figura 33.</i> Opciones de viaje provistas por la librería de Android.....	61
<i>Figura 34.</i> Herramienta Apache JMeter configuración de hilos.....	64
<i>Figura 35.</i> Herramienta Apache JMeter Configuración de petición HTTP.....	65
<i>Figura 36.</i> Herramienta Apache JMeter configuracionVista de resultados.....	66
<i>Figura 37.</i> Herramienta Apache JMeter resultados con 10 muestra.....	67
<i>Figura 38.</i> Herramienta Apache JMeter Árbol de resultados 10 muestras.....	67
<i>Figura 39.</i> Herramienta Apache JMeter resultado con 100 muestras.....	68
<i>Figura 40.</i> Herramienta Apache JMeter resultado 1000 muestras	69

ÍNDICE DE ANEXOS

Anexo 1. Manual técnico de despliegue del sistema de visualización de obras Salesiana	76
--	----

RESUMEN

El proyecto “Análisis, diseño e implementación de un geo portal orientado a dispositivos móviles en Sistema Operativo Android para visualizar las obras Salesianas en el sector Educativo, Pastoral y Auto gestionado”, facilita el acceso a la información de cada una de las casas Salesianas para visualizar su ubicación e información en Google Maps a través del escaneo de códigos QR(quick response code, «código de respuesta rápida») asignados a cada casa salesiana. La aplicación está basado en el estudio e investigación de los códigos QR sobre la plataforma Google Android para la implementación en aplicaciones móviles, puede ser ejecutada por una persona (usuario) que disponga de un dispositivo móvil con sistema operativo Android y la aplicación instalada (visualizador de obras Salesianas), adicionalmente debe poseer conexión a internet.

El sistema también sugiere las posibles rutas a tomar en base al cálculo de la posición actual en relación al destino especificado por el código QR, adicionalmente nos provee el tiempo estimado de viaje según la ruta y la opción de viaje (ejemplo caminar o en auto). En el lado del servidor se tiene un administrador web para gestionar los permisos, perfiles de los usuarios, administrar los códigos QR (crear, eliminar) y buscar las casas Salesianas por el nombre.

ABSTRACT

The project "Analysis, design and implementation of a geoportal to mobile devices that support Android Operating System in order to visualize the Salesian work in the Educative, Pastoral and Self-managed fields" facilitates the information access of each one of the Salesian houses to visualize its location and its information on Google Maps. This is achieved through the scanning of QR codes (Quick Response codes) assigned to each Salesian house. The application is based on the study and research of the QR codes on the Google Android platform for implementing mobile applications, it can be executed by a person (user) who has a mobile device with Android OS and the installed application (Salesian work visualizer), aside from an Internet connection.

The system also suggests possible routes to take based on the calculation of the current position in relation to the destination specified by the QR code; additionally it offers the estimated travel time by route and travel option (e.g. walking or car). Regarding the server, it has a web administrator to manage permissions, user profiles, QR Codes (create, delete); and to search Salesian houses by name.

INTRODUCCIÓN

El proyecto “Análisis, diseño e implementación de un geo portal orientado a dispositivos móviles en Sistema Operativo Android para visualizar las obras Salesianas en el sector Educativo, Pastoral y Auto gestionado”, tiene como objetivo facilitar el acceso a la información de obras Salesianas a través de dispositivos móviles.

El incremento de teléfonos inteligentes a nivel mundial que utilizan sistema operativo Android hace que se facilite la utilización de esta tecnología para encontrar o localizar sitios determinados en forma rápida y precisa en un mapa. La aplicación (visualizador de obras Salesianas) permite visualizar de una manera rápida la ubicación de los diferentes lugares donde existen obras Salesianas sin necesidad de acceder a un portal o sub menú, lo hace a través del escaneo de un código QR (quick response code, «código de respuesta rápida»), a la cual se puede acceder desde cualquier lugar siempre que se disponga de conexión a internet y debe tener instalado en el dispositivo móvil el visualizador de obras Salesianas.

Se desarrolló la aplicación (visualizador de obras Salesianas) utilizando Eclipse (Entorno de desarrollo) para la programación del módulo de usuario final y PHP (Entorno de desarrollo) para el módulo de administración. En el desarrollo de esta aplicación se investigó y analizó las diferentes librerías como son ZXing, Google Android, Google Maps y Android API v2.

La organización de este trabajo se presenta en cinco capítulos que se detallan a continuación

El capítulo uno “Marco teórico”, contiene información de la metodología XtremeProgramming (XP) y herramientas de desarrollo utilizadas.

El capítulo dos “Requerimientos del sistema” presenta los elementos necesarios para el sistema desarrollado.

El capítulo tres “Diseño” parte del análisis de los requerimientos para crear los diagramas de clases, base de datos, secuencia, entre otros que son insumos para la creación del sistema.

El capítulo cuatro “Desarrollo” presenta la creación de la aplicación en base al diseño.

El capítulo cinco “Implementación y pruebas” se realizaron pruebas de stress y funcionalidad al sistema para obtener un sistema confiable.

Por último se presenta las conclusiones, recomendaciones y se incluyen los anexos obtenidos en el desarrollo del visualizador de obras Salesianas.

CAPÍTULO 1

JUSTIFICACIÓN Y FORMULACIÓN DEL PROBLEMA

1.1 Justificación

Actualmente no se tiene fácil acceso a la información de obras Salesianas a través de dispositivos móviles, por lo cual se ha visto necesario desarrollar un visualizador de obras Salesianas.

El visualizador de obras Salesianas está orientado a dispositivos móviles, que permite aprovechar las librerías disponibles de Android como es google map, google play, de manera fácil y sencilla, lo que facilita la integración en nuestro aplicativo obteniendo un producto funcional, para que muchos usuarios se informen y conozcan las casas Salesianas distribuidas a nivel nacional mediante sus dispositivos móviles .

1.2 Formulación del problema

“La congregación salesiana fundada en 1859 cuya principal misión fue la educación de los jóvenes más necesitados. Llega al Ecuador en el año de 1888 por decreto del presidente José M. Plácido Caamaño, actualmente los salesianos están distribuidos en 26 comunidades en costa sierra y amazonia”. (A Bartoli, 2002, pág. 171)

No existe un aplicativo que permita visualizar las obras Salesianas usando códigos QR en dispositivos móviles con sistema operativo Android en la Universidad Politécnica Salesiana Quito. Por lo que no se aprovecha al máximo la información disponible en el geo portal de la comunidad salesiana, por lo que se desconoce la información y propósito de las obras Salesianas existentes en las diferentes provincias del Ecuador su ubicación y las posibles rutas para llegar.

El contar con una aplicación la cual nos permita acceder a la información de las casas Salesianas de una manera rápida a través de dispositivos móviles para familiarizarnos con las obras las cuales han sido implementadas durante la llegada de las congregaciones fundadas por Juan Don Bosco es de mucha ayuda e importancia para toda la Comunidad Salesiana y su vinculación con la sociedad.

1.3 Importancia

En la actualidad el mercado de dispositivos móviles con sistema operativo Android y acceso a Internet es muy extenso, es fácil encontrar dispositivos que posean estas características, tienen facilidad al acceso del público, los precios varían de acuerdo al equipo y marca. Por tal motivo al implementar esta aplicación queremos permitir y dar a conocer las obras Salesianas en el país mediante los dispositivos móviles lo que será de mucha ayuda, para poder unir más comunidades Salesianas al servicio de la sociedad.

1.4 Objetivos

1.4.1 Objetivo general.

Analizar diseñar e implementar un geo portal en sistema operativo Android orientado a dispositivos móviles

1.4.2 Objetivos específicos.

- Utilizar el código QR¹, el mismo que facilita el acceso a google maps en función a la georreferenciación provista por el código QR.
- Mostrar información específica y detallada de una obra salesiana extraída de la base de datos del portal.
- Visualizar las obras Salesianas con el uso de la tecnología móvil.
- Visualizar las posibles rutas para llegar a las obras Salesianas especificadas.

1.5 Metodología

La metodología utilizada en el desarrollo de la propuesta para el visualizador de obras Salesianas en sistema operativo Android orientado a dispositivos móviles, es Extreme Programing (XP).

¹ QR (quick response code, «código de respuesta rápida)

1.5.1 XtremeProgramming (XP)

Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP(Xtreme Programming) se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP(Xtreme Programming) se define como adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. (Bolivariana, 2015, pág. 1)

1.5.2 Metodología de Programación Extrema o XP

Clasificada como un tipo de metodología ágil, se enfoca principalmente al enfoque orientado a objetos, según Pressman (2010) engloba un conjunto de reglas y prácticas que ocurren en el contexto de cuatro actividades estructurales, las cuales puede observarse en la figura1. (Rojas, 2013, p. 1)

Etapas de XP

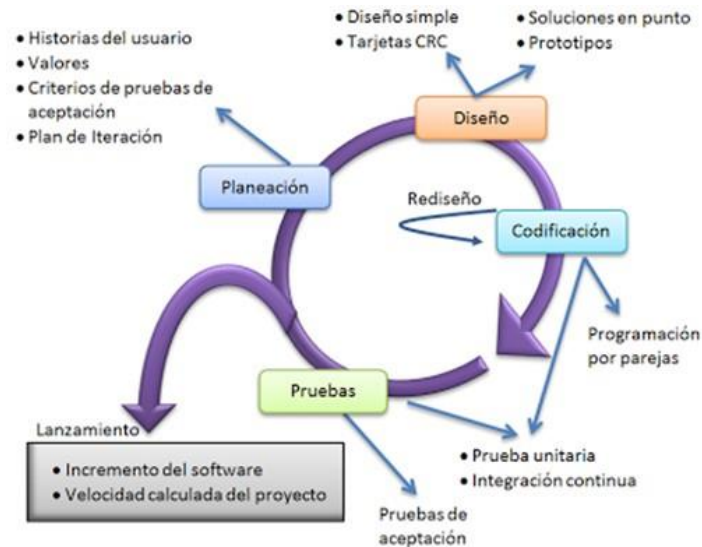


Figura 1. Etapas de XP

Fuente Code Jobs

1.5.2.1 Planeación

Se escucha los requerimientos, las necesidades de los usuarios, esta actividad conlleva a la creación de historias o historias de usuario, las cuales describen la salida necesaria, características y funcionalidades del software a elaborar, cada historia es colocada en una tarjeta indizada, así el cliente asigna un valor o prioridad a determinada historia, tomando como referencia el valor, que la actividad descrita en la tarjeta representa para el negocio. Posteriormente el equipo de trabajo evalúa cada historia y se le asigna un costo medido en semanas de desarrollo, en el caso que este sea superior a tres semanas, se solicita al usuario que descomponga la historia original en otras más pequeñas y que puedan ser abarcadas en dicho tiempo. Una vez llegado a un compromiso sobre las entregas, las historias son ordenadas según su desarrollo de la siguiente manera

- Todas las historias se implementarán de inmediato.
- Las historias con más valor entrarán en la programación de actividades y se implementarán en primer lugar.
- Las historias más riesgosas formarán parte de las actividades que se implementarán en primer lugar.

Una vez realizada la primera entrega de software (o incremento), se calcula la velocidad del proyecto. Este valor permite Facilitar la estimación de las fechas de entrega y programar actividades para entregas posteriores, determinar el grado de compromiso para todas las historias durante el desarrollo del proyecto, ajustando la velocidad de las entregas.

A medida que se ejecuta el proyecto, el cliente puede agregar más historias, cambiar el valor de una ya existente, descomponerla o eliminarlas, así el equipo de desarrolladores modifica sus planes y reconsidera las entregas faltantes.

1.5.2.2 Diseño

Se le da preferencia a la sencillez sobre representaciones más complejas, así el diseño guía la implementación de una historia mientras se escribe, se inhibe de agregar funcionalidad adicional, ya que dicha funcionalidad será agregada después.

Se utilizan las tarjetas CRC (Clase, Responsabilidad, Colaborador), para identificar y organizar las clases orientadas a objetos.

En aquellos casos donde se encuentre un problema de diseño difícil, se recomienda la creación de un prototipo operativo para esa porción de diseño, implementando y evaluando el prototipo de diseño, así cuando se implemente la versión final, se disminuyen los riesgos.

La programación extrema estimula el rediseño, así el comportamiento externo del código queda inalterable, pese a las mejoras realizadas en su estructura interna, al modificar y simplificar el diseño interno, esto sólo se puede alcanzar gracias a las ventajas que ofrece la encapsulación y el polimorfismo de la programación orientada a objetos.

1.5.2.3 Codificación

El primer paso de la codificación, no es escribir el código mismo del desarrollo, al contrario, se elaboran las pruebas unitarias de cada una de las entregas relacionadas con las historias, de esta manera el desarrollador centrará sus esfuerzos de programación en pasar las pruebas previamente elaboradas, es decir, las pruebas actúan como guías de acción para el programador.

Se recomienda que la codificación sea realizada en parejas sobre una misma estación de trabajo, esto mejora notablemente el desempeño de los programadores y la calidad del producto, el resultado de este equipo de trabajo se va integrando con el de los demás. En ocasiones esta labor deberá ser llevada a cabo por un equipo de integración, que junte en un mismo espacio el producto resultante de las diversas parejas de programadores.

1.5.2.4 Pruebas

Las pruebas unitarias, elaboradas como primer paso en la etapa de la codificación, se implementan con una estructura que permita su automatización, para que así puedan implementarse repetidas veces y con facilidad. Dichas pruebas pueden ejecutarse a diario con el objeto de corregir cualquier desviación o anomalía del software a tiempo.

También se requiere ejecutar pruebas de aceptación por parte del cliente, estas pruebas se centran en las características y funcionalidades generales del sistema y que son visibles y auditables por parte del cliente. (Rojas, 2013, p. 1)

1.6 Alcance

El producto final permitirá satisfacer necesidades de la visualización de las casas Salesianas de una manera ágil a través de la utilización de los códigos QR (quick response code, «código de respuesta rápida») en la gestión de la visualización a través de dispositivos móviles que usen sistemas Android y el resultado será un aplicativo totalmente funcional sobre una plataforma de Android , con requerimientos establecidos por la comunidad Salesiana, el cual estará formado de los siguientes módulos.

1.6.1 Módulos servidor

Permite la creación, edición y eliminación de usuarios, perfiles de Usuario y permisos que asignados en el sistema. Agregar, editar, eliminar y consultar códigos QR asignados a las casas Salesianas. Se diseñó el sistema para que guarde la información generada de los códigos QR en la base de datos

Módulos cliente web

Permite la consulta de los códigos asignados a cada casa Salesiana y además la posibilidad de descargar el visualizador para la instalación en su dispositivo móvil

Módulo cliente Android

Llevará a cabo la función de buscar las casas Salesianas a través del escaneo de un código QR y mostrarlo en google map.

CAPÍTULO 2

MARCO TEÓRICO

2.1 Google Android

Es un sistema operativo de Licencia Libre y de Código Abierto, basado en Linux (kernel de Linus Torvalds); provee de gran estabilidad y seguridad. Tiene gran variedad de hardware no solo en teléfonos móviles sino también en Tablets y Netbooks.

Android es un sistema operativo con multitarea, permite ejecutar varias aplicaciones al mismo tiempo. Los desarrolladores que vienen de una plataforma diferente pueden encontrar un cambio llamativo al aplicar este concepto. (libre, 2011, pág. 1)

2.1.1 Versiones Android

Para los que están entrando a conocer el mundo Android, deben saber que posee varias versiones, es decir al igual que Windows, este va mejorando para brindar mayor fluidez y una experiencia de usuario cada vez mejor y moderna. Google mejora su software constantemente para ofrecer al usuario diferentes versiones de los cuales son compatibles con algunos dispositivos de mayor antigüedad.

Esto nos indica por qué se presentan los constantes desarrollos en el software Android, que a diferencia de Apple, realiza cambios constantes y que en su mayoría son visibles a nivel de fluidez e interfaz.

A continuación se detalla las versiones Android que desde sus inicios en el año 2008 han sido desarrolladas hasta la fecha (C, 2013, pág. 1)

Tabla 1.
Versiones de Android

VERSIÓN ANDROID	NOMBRE VERSIÓN	FECHA LANZAMIENTO
1.0	Apple Pie	23 de septiembre de 2008
1.1	Banana Bread	9 de febrero de 2009
1.5	Cupcake	30 de abril de 2009
1.6	Donut	15 de septiembre de 2009
2.0/2.1	Eclair	26 de octubre de 2009
2.2.x	Froyo	20 de mayo de 2010
2.3.x	Gingerbread	6 de diciembre de 2010
3.x	Honeycomb	22 de febrero de 2011
4.0.x	Ice Cream Sandwich	19 de octubre de 2011
4.1 - 4.2	Jelly Bean	13 de noviembre de 2012
5.0	Key Lime Pie	En desarrollo

Nota. El desarrollo está soportado desde la versión 2.2.x Froyo lanzado el 20 de mayo de 2010

Elaborado por Hugo Cajamarca

2.2 Códigos QR (quick response code, «código de respuesta rápida»).

Los códigos QR, son un tipo de códigos de barras bidimensionales, a diferencia de un código de barras convencional (por ejemplo EAN-13, Código 3 de 9, UPC), la información está codificada dentro de un cuadrado, permitiendo almacenar gran cantidad de información alfanumérica. Los códigos QR son fácilmente identificables por su forma cuadrada y por los tres cuadros ubicados en las esquinas superiores e inferior izquierda ver en la Figura 2. (QR, Codigos QR, 2015, pág. 1)

Código QR



Figura 2. Código QR

Fuente (QR, Codigos QR, 2015)

2.2.1 Para qué sirve un código QR

Aunque el desarrollo inicial de los códigos QR tenía como objetivo principal su utilización en la industria de la automoción, hoy por hoy la posibilidad de leer códigos QR desde teléfonos y dispositivos móviles permite el uso de Qr Codes en un sin fin de aplicaciones completamente diferentes de las que originales como pueden ser

- Publicidad
- Campañas de marketing
- Merchandising
- Diseño gráfico
- Papelería corporativa (tarjetas de visita, catálogos)
- Internet, Webs, blogs

2.2.2 Qué puede contener un código QR

Un código QR es una manera fácil y sencilla de interactuar con un dispositivo móvil y permitir realizar acciones automáticamente con el terminal como por ejemplo

- Abrir la URL de una página Web o perfil social
- Leer un texto
- Enviar un email
- Enviar un SMS
- Realizar un llamada telefónica
- Guardar un evento en la agenda
- Ubicar una posición geográfica en un google maps. (QR, Codigos QR, 2015, pág. 1)

2.3 ZXing

ZXing es una librería procesadora de imágenes multi-formato en 1D/2D y de código abierto. Actualmente es capaz de reconocer los formatos UPC-A, UPC-E, EAN-8, EAN-13, Códigos 39, 93, 128, ITF, Codabar, RSS-14 (en todas sus variantes), Matriz de datos (Data Matrix), Aztec, PDF 417 y por supuesto los populares códigos QR, muy usados desde que la tecnología móvil está presente en nuestras vidas. (Alvares, ZOMWI informatica y algo mas., 2013, pág. 1)

2.4 Apache Tomcat

Apache, también conocido como simplemente Tomcat o Jakarta Tomcat, es un servidor web multiplataforma que funciona como contenedor de servlets y que se desarrolla bajo el proyecto denominado Jackarta perteneciente a la Apache Software Foundation bajo la licencia Apache 2.0 y que implementa las especificaciones de los servlets y de JavaServer Pages o JSP de Sun Microsystem. Dicho servidor es mantenido y desarrollado por miembros de la fundación y voluntarios independientes, los cuales tienen libre acceso al código fuente bajo los términos establecidos por la Apache Software Foundation. (foundation, Apache Tomcat, 2014, pág. 1)

2.4.1 Historia

Apache Tomcat comenzó siendo una implementación de servlets iniciada por James Duncan Davidson, que trabajaba como arquitecto software en Sun Microsystem y que posteriormente ayudó al proyecto de código abierto. Duncan, inicialmente pensó que el proyecto se convertiría en software de código abierto y además quiso ponerle un nombre de animal, en este caso Tomcat (gato) ya que, de algún modo pretendía trasladar la posibilidad de cuidarse por sí mismo, es decir, de ser independiente. Las primeras distribuciones de Apache Tomcat fueron las 3.0.x aunque las versiones estables más recientes son las 6.0.30 y la 7.0.6 que implementan las especificaciones de Servlet 2.5 y JSP 2.1. (foundation, Apache Tomcat, 2014, pág. 1)

2.4.2 Estructura

La jerarquía de directorios de Apache Tomcat es la siguiente

- ***bin*** arranque, cierre, scripts y ejecutables.
- ***common*** clases comunes que puede utilizar Catalina (contenedor de servlets) y las aplicaciones web.
- ***conf*** ficheros XML y la correspondiente DTD para la configuración de Apache Tomcat.
- ***logs*** logs del contenedor de servlets y de las aplicaciones.
- ***server*** clases usadas por el contenedor de servlets.
- ***shared*** clases compartidas por todas las aplicaciones web.
- ***webapps*** directorio que contiene las aplicaciones web.
- ***work*** almacenamiento temporal de ficheros y directorios. (foundation, Apache Tomcat, 2014, pág. 1)

2.4.3 Componentes

A partir de la versión 4.x Tomcat fue lanzado con el contenedor de servlets "Catalina", con el contenedor HTTP "Coyote" y un motor para JSP llamado "Jasper". Las principales características de estos tres componentes son

Catalina

Dicho componente implementa las especificaciones de servlets y JSP. Para Apache Tomcat el elemento principal es una base de datos de nombres de usuarios, password y roles, permitiendo a Catalina integrarse donde ya existe información de autenticación como describe la especificación de servlets.

Coyote

Componente conector que admite el protocolo HTTP 1.1 para el servidor web y que escucha en un puerto TCP especificado por el servidor y envía la solicitud al motor Tomcat para que éste procese la solicitud y envíe una respuesta al cliente.

Jasper

Jasper analiza archivos JSP para compilar el código Java y, si se producen cambios, éste los vuelve a compilar. Desde la versión 5.x se usa Jasper 2 que es JavaServer para webs 2.0. Jasper 2 ha introducido las siguientes novedades

- Recompilación al producirse un cambio.
- Incorpora el compilador JDT de Java.
- Puesta en común de etiquetas JSP. (foundation, Apache Tomcat, 2014, pág. 1)

2.4.4 Características y evolución

Como se ha dicho anteriormente, Apache Tomcat es un servidor web que da soporte a servlets y JSPs de modo que, no es un servidor de aplicaciones. Dado que dicho producto fue desarrollado en Java, éste puede ejecutarse sobre cualquier sistema operativo, previa instalación de la máquina virtual de Java aunque, también se puede usar con MAMPP (Mac OS X), LAMPP (GNU/Linux), WAMPP (Windows) o XAMPP (cualquier sistema operativo). Además, puede funcionar como servidor web por sí mismo sin embargo, en sus inicios se pensaba que dicho servidor era recomendable usarse en entornos de desarrollo con requisitos mínimos de velocidad. En la actualidad no existe esta percepción y por esto, es usado como servidor web autónomo en entornos con alto nivel de tráfico y alta disponibilidad.

Más en concreto, las principales características que soporta la última versión estable son

- Autenticación de acceso básico.
- Negociación de credenciales.
- HTTPS.
- Alojamiento compartido.
- CGI o interfaz de entrada común.
- Servlets de Java.
- SSL.
- Consola de administrador.

La evolución del producto y sus principales características de cada una de las versiones han sido

- Implementación a partir de Servlet 2.2 y JSP 1.1.
- Recarga de servlets.
- Funciones básicas de HTTP.Tomcat 4.x.
- Implementado a partir de las especificaciones Servlet 2.3 y JSP 1.2.
- Contenedor de servlets rediseñado como Catalina.
- Motor JSP rediseñado con Jasper.
- Conector Coyote.
- Java Management Extensions (JMX), JSP y administración basada en Struts. Tomcat 5.x.
- Implementado a partir de las especificaciones Servlet 2.4 y JSP 2.0.
- Recolección de basura reducida.
- Capa envolvente nativa para Windows y Unix para la integración de las plataformas.
- Análisis rápido JSP.Tomcat 6.x.
- Implementado de Servlet 2.5 y JSP 2.1.
- Soporte para Unified Expression Language 2.1.
- Diseñado para funcionar en Java SE 5.0 y posteriores.
- Soporte para Comet a través de la interfaz CometProcessor. Tomcat 7.x.
- Implementado de Servlet 3.0 JSP 2.2 y EL 2.2.
- Mejoras para detectar y prevenir "fugas de memoria" en las aplicaciones web.

- Limpieza interna de código.
- Soporte para la inclusión de contenidos externos directamente en una aplicación web. (foundation, Apache Tomcat, 2014, pág. 1)

2.5 PostgreSQL 9

Es un sistema de gestión de base de datos relacional orientada a objetos y libre, publicado bajo la licencia BSD. Es más completo que MySQL ya que permite métodos almacenados, restricciones de integridad, vistas, etc. Como muchos otros proyectos de código abierto, el desarrollo de PostgreSQL no es manejado por una sola empresa sino que es dirigido por una comunidad de desarrolladores y organizaciones comerciales las cuales trabajan en su desarrollo. Dicha comunidad es denominada el PGDG (PostgreSQL Global Development Group). Es el sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones no tiene nada que envidiarle a otras bases de datos comerciales. Utiliza el lenguaje SQL para llevar a cabo sus búsquedas de información, las bases de datos generadas dentro de servidores de SQL son bases de datos relacionales. Utiliza un modelo cliente/servidor y usa Multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando. (PostgreSQL, PostgreSQL, 2013, pág. 1)

2.6 PostGIS 1.5

PostGIS Es una extensión al sistema de base de datos objeto-relacional PostgreSQL.

Permite el uso de objetos *GIS* (Geographic information systems). PostGIS incluye soporte para índices GiST basados en R-Tree, y funciones básicas para el análisis de objetos GIS.

Esta creado por Refrations Research Inc, como un proyecto de investigación de tecnologías de bases de datos espaciales. Está publicado bajo licencia GNU.

Con PostGIS se puede usar todos los objetos que aparecen en la especificación OpenGIS como puntos, líneas, polígonos, multilíneas, multipuntos, y colecciones geométricas.

Objetos GIS.

Los objetos GIS soportados por PostGIS son de características simples definidas por OpenGIS. Actualmente PostGIS soporta las características y el API de representación de la especificación OpenGIS pero no tiene varios de los operadores de comparación y convolución de esta especificación.

Ejemplos de la representación en modo texto

- POINT(0 0 0)
- LINESTRING(0 0,1 1,1 2)
- POLYGON((0 0 0,4 0 0,4 4 0,0 4 0,0 0 0),(1 1 0,2 1 0,2 2 0,1 2 0,1 1 0))
- MULTIPOINT(0 0 0,1 2 1)
- MULTILINESTRING((0 0 0,1 1 0,1 2 1),(2 3 1,3 2 1,5 4 1))
- MULTIPOLYGON(((0 0 0,4 0 0,4 4 0,0 4 0,0 0 0),(1 1 0,2 1 0,2 2 0,1 2 0,1 1 0)),((-1 -1 0,-1 -2 0,-2 -2 0,-2 -1 0,-1 -1 0)))
- GEOMETRYCOLLECTION(POINT(2 3 9),LINESTRING((2 3 4,3 4 5))

En los ejemplos se pueden ver características con coordenadas de 2D y 3D (ambas son permitidas por PostGIS). Podemos usar las funciones force_2d() y force_3d() para convertir una característica a 3d o 2d. (Postgis.refrations.net, 2015, pág. 4)

2.7 Quantum GIS (QGIS) 1.8

QGIS (antes Quantum GIS) es un programa de código abierto y gratuito que sirve para visualizar y procesar datos geográficos. QGIS funciona en diferentes sistemas operativos, tales como Windows, Linux y Mac. Este programa provee herramientas básicas para poder hacer las labores de visualización, recopilación de información geográfica, y análisis de geodatos (información geográfica), además de impresión de mapas. (Foundation, Quantum GIS, 2013, pág. 1).

Las principales características son Crea, edita, visualiza, analiza y publica información geoespacial, ya sea para Windows, Mac, Linux, BSD y Android. Navega y pre visualiza datos y metadatos. Herramientas de Análisis Espacial. Importación y exportación de datos GPS. (GIS, 2013, pág. 1)

2.8 Eclipse 4.2

Eclipse es una plataforma de desarrollo, diseñada para ser extendida de forma indefinida a través de plug-ins. Fue concebida desde sus orígenes para convertirse en una plataforma de integración de herramientas de desarrollo. No tiene en mente un lenguaje específico, sino que es un IDE genérico, aunque goza de mucha popularidad entre la comunidad de desarrolladores del lenguaje Java. (dev, 2014, pág. 1)

2.9 Google Maps Android API v2

Google Maps proporciona un servicio de cartografía online que se puede integrar en aplicaciones Android. Estudiaremos la versión 2 del API incorpora interesantes ventajas respecto a la versión anterior. Entre estas ventajas destaca el menor tráfico intercambiado con el servidor, la utilización de fragments y los gráficos en 3D. Como inconveniente resaltar que la nueva versión solo funciona en el dispositivo con Google Play instalado. (Valencia, 2015, p. 1)

2.10 Phpqrcode

PHP Código QR es de código abierto (LGPL), biblioteca para la generación de código QR , código de barras de 2 dimensiones . Basado en libqrencode biblioteca C , proporciona API para crear Código QR imágenes de código de barras (PNG , JPEG gracias a GD2) . Implementado puramente en PHP, sin dependencias externas (excepto GD2 si es necesario).

Algunas de las características de la biblioteca incluye

- Soporta las versiones de código QR (tamaño) 1-40.
- Numérico, alfanumérico, 8 bits y Kanji codificación.
- Implementadas puramente en PHP, sin dependencias externas excepto GD2.
- Las exportaciones a PNG, imágenes JPEG, también exporta como bit-tabla TCPDF 2-D integración API de código de barras.
- Fácil de configurar.
- Caché de datos para el cálculo de aceleración.
- Volcado de datos de depuración, el registro de errores, la evaluación comparativa tiempo. (PHPQRCODE, s.f., pág. 1)

2.11 Bootstrap

“Es una herramienta Open Source para el desarrollo rápido de aplicaciones web que ha ido creciendo en popularidad hasta convertirse en uno de los proyectos más destacados en la plataforma de código abierto GitHub”

2.11.1 Accesibilidad

Bootstrap 3 utiliza los estándares web de tal forma que con muy poco esfuerzo se pueden crear sitios web accesibles para personas que utilizan tecnologías de apoyo para navegar.

2.11.2 La licencia de Bootstrap

Bootstrap se publica bajo la licencia "Apache 2 License" y está protegido por el siguiente copyright "copyright 2013 Twitter". Para comprender en detalle lo que puedes y no puedes hacer con esta licencia, lee las siguientes secciones.

2.11.2.1 Lo que te permite la licencia

- Descargar y usar gratuitamente Bootstrap entero o alguna de sus partes, tanto para uso personal, empresarial y/o comercial.
- Utilizar e incluir Bootstrap en cualquier paquete o distribución de software que crees.

2.11.2.2 Lo que te prohíbe la licencia

- Redistribuir cualquier parte de Bootstrap sin mencionar quiénes son sus autores.
- Utilizar cualquier elemento propiedad de la empresa Twitter de forma que parezca que Twitter apoya o tiene alguna relación con tu proyecto.
- Utilizar cualquier elemento propiedad de la empresa Twitter de forma que parezca que tú has creado ese elemento de Twitter. (desarrolloweb, 2013, pág. 1)

2.12 Apache-JMeter-2.11

Apache JMeter es una herramienta de carga diseñada para realizar pruebas de rendimiento y pruebas funcionales sobre aplicaciones web. Desarrollado por THE APACHE SOFTWARE FOUNDATION, la primera versión (v1.0) data de marzo del 2001, mientras que versión actualmente homologada en EJIE es la Originalmente el Apache JMeter fue diseñado para realizar pruebas de estrés sobre aplicaciones web (pruebas web clásicas). Sin embargo hoy en día su arquitectura ha evolucionado, ahora no sólo puede llevar a cabo pruebas en componentes típicos de Internet (HTTP), sino también puede realizar pruebas sobre Bases de Datos, scripts Perl, servlets, objetos java, servidores FTP y prácticamente cualquier

medio de los que se pueden encontrar en la red. (jakarta.apache.org, 2014, pág. 1)

2.13 Google Maps

Es el nombre de un servicio gratuito de Google. Es un servidor de aplicaciones de mapas en Web. Ofrece imágenes de mapas desplazables, así como fotos satelitales del mundo entero e incluso la ruta entre diferentes ubicaciones. Desde el 6 de octubre del 2005, Google Maps es parte de Google Local. (googlemaps-nabil.blogspot.com, 2014, pág. 1)

2.13.1 Desarrollo

Google Maps fue anunciado por primera vez en Google Blog el 8 de febrero de 2005. Originalmente soportaría solo a los usuarios de Internet Explorer y Mozilla Firefox, pero el soporte para Opera y Safari fue agregado el 25 de febrero de 2005. El software estuvo en su fase beta durante seis meses, antes de convertirse en parte de Google Local, el 6 de octubre de 2005. Como en las aplicaciones web de Google, se usan un gran número de archivos Javascript para crear Google Maps. Como el usuario puede mover el mapa, la visualización del mismo se baja desde el servidor. Cuando un usuario busca un negocio, la ubicación es marcada por un indicador en forma de pin, el cual es una imagen PNG transparente sobre el mapa. Para lograr la conectividad sin sincronía con el servidor, Google aplicó el uso de AJAX dentro de esta aplicación. Es una aplicación para el desarrollo de mapas. Unos días más tarde Google fue el buscador de internet más famoso. (googlemaps-nabil.blogspot.com, 2014, pág. 1)

2.13.2 Básicas

Google Maps ofrece la capacidad de realizar acercamientos y alejamientos para mostrar el mapa. El usuario puede controlar el mapa con el mouse o las teclas de dirección para moverse a la ubicación que se desee. Para permitir un movimiento más rápido, las teclas "+" y "-" pueden ser usadas para controlar el nivel de zoom. Los usuarios pueden ingresar una dirección, una intersección o un área en general para buscar en el mapa.

Los resultados de la búsqueda pueden ser restringidos a una zona, gracias a Google Local. Por ejemplo, si alguien quiere consultar por "Waffles in Ottawa" (en español, gofres en Ottawa) para encontrar restaurantes que sirven ese plato cerca de la ciudad. Las búsquedas pueden encontrar una amplia gama de restaurantes, hoteles, teatros y negocios generales. Como otros servicios de mapa, Google Maps permite la creación de pasos para llegar a alguna dirección. Esto permite al usuario crear una lista paso a paso para saber cómo llegar a su destino, calculando el tiempo necesario y la distancia recorrida entre las ubicaciones. Los desarrolladores pueden aplicar nuevas formas de ver el mundo gracias a estas APIS. (googlemaps-nabil.blogspot.com, 2014, pág. 1)

2.14 Google Play Services

Es una aplicación del sistema de Android que permite tener el resto de aplicaciones de nuestro terminal siempre actualizadas, ya que se encargará de comprobar que todas las apps instaladas están en la última versión disponible.

Las funciones principales de Google Play Services son la autenticación de servicios de Google, la sincronización de contactos, el acceso a la última configuración de privacidad del usuario, y los servicios basados en la ubicación de mayor calidad y menor potencia.

Además de todo esto, Google Play Services mejora la experiencia general de uso del terminal. Permite agilizar las búsquedas sin conexión, proporciona mapas más envolventes y mejora la experiencia de juego mediante la optimización de memoria RAM

Google Play Services es una aplicación realmente imprescindible mientras se utiliza un terminal con sistema operativo Android, ya que sin ella instalada es posible que muchas otras aplicaciones comiencen a sufrir problemas. (googlemaps-nabil.blogspot.com, 2014, pág. 1)

2.15 Las pruebas y el desarrollo de software

La etapa de pruebas es una de las fases del ciclo de vida de los proyectos. Se la podría ubicar después del análisis, el diseño y la programación, pero dependiendo del proyecto en cuestión y del modelo de proceso elegido, su realización podría ser en forma paralela a las fases citadas o inclusive repetirse varias veces durante la duración del proyecto.

La importancia de esta fase será mayor o menor según las características del sistema desarrollado, llegando a ser vital en sistemas de tiempo real u otros en los que los errores sean irrecuperables. Las pruebas no tienen el objeto de prevenir errores sino de detectarlos. Se efectúan sobre el trabajo realizado y se deben encarar con la intención de descubrir la mayor cantidad de errores posible. Para realizar las pruebas se requiere gente que disfrute encontrando errores, por eso no es bueno que sea el mismo equipo de desarrollo el que lleve a cabo este trabajo. Además, es un principio fundamental de las auditorías. (Pablo Suarez, 2003, pág. 5)

En un escenario de proyecto ideal se podría generar casos de prueba para cubrir todas las posibles entradas y todas las posibles situaciones por las que podría atravesar el sistema de software. Pero hay un gran problema en esto el número de casos de prueba para un sistema complejo sería tan

grande que no alcanzaría una vida para terminar con las pruebas. Como producto de esto nadie realiza una prueba exhaustiva. Las pruebas cuestan mucho dinero por lo que existe una máxima “pague por la prueba ahora o pague el doble por el mantenimiento después”. Esto conlleva a planificar bien las pruebas con anticipación y determinar los resultados que se desea obtener. (Pablo Suarez, 2003, pág. 7)

CAPÍTULO 3

DISEÑO

El visualizador de obras Salesianas, permitirá el fácil acceso a la ubicación de las casas Salesianas a través de dispositivos móviles, administración de usuarios, creación de perfiles que tendrá un único control de autenticación, autorización de accesos al geo portal.

Éste visualizador de obras Salesianas dispone de un sistema de gestión de perfiles y credenciales con los cuales se podrá establecer políticas de navegación en el geo portal, permitiendo el acceso a los recursos necesarios del sistema que se definen de acuerdo a la credencial proporcionada al usuario.

3.1 Especificación de requisitos del Software

La especificación de requisitos del software comprende las necesidades funcionales de tener acceso a las páginas que tiene el proyecto para realizar actividades de acuerdo al rol y perfil que se asignará a cada usuario con su respectiva descripción detallada. Finalmente se incluye los productos de la metodología XP, específicamente las historias de usuarios y tarjetas CRC. Con toda esta información reunida se obtiene una idea clara de los requisitos necesarios en el sistema.

Tabla 2.

Roles de los usuarios en el sistema

Usuario	Funciones	Requisitos mínimos
Administrador	Es el encargado de la administración del sistema como crear, editar , eliminar, códigos QR.	.Conocimientos intermedios computación .Conocimiento de base de datos .Conocimientos administración portales web
Invitado	Consultar los códigos QR generados	.Conocimientos básicos de computación

Nota. Invitado=usuario con acceso restringido. Fuente Autor

Elaborado por Hugo Cajamarca

Tarjetas CRC (Clase-Responsabilidad-Colaboración)

Las tarjetas CRC (Clase-Responsabilidad-Colaboración) son una herramienta de brainstorming, usado como metodología para el diseño de software orientado a objetos creada por Kent Beck y Ward Cunningham. Es una técnica para la representación de sistemas orientados a objetos. Son un puente de comunicación entre diferentes participantes.

Las tarjetas CRC, permiten representar un sistema orientado a objetos. En estas se establecen, como una clase colabora con otras clases para cumplir sus responsabilidades y asociaciones entre clases. Todo este proceso permitirá delinear el sistema a implementar ver en las siguientes tablas.

Tabla 3.

Tarjeta CRC de la clase usuario

Nombre de la Clase: Usuario	
Descripción: Esta clase se encarga de todo lo relacionado a la gestión de usuarios y sus características.	
Responsabilidades <ul style="list-style-type: none">• Crear Usuario• Actualizar Usuario• Consulta Usuario• Eliminar Usuario• Validar Acceso• Cambiar clave	Colaboradores <ul style="list-style-type: none">• Perfil

Nota. Autor

Elaborado por Hugo Cajamarca

Tabla 4.
Tarjeta CRC de la clase perfil

Nombre de la Clase: Perfil	
Descripción: Esta clase contiene características de los perfiles de usuario del sistema	
Responsabilidades <ul style="list-style-type: none"> • Crear Perfil • Actualizar Perfil • Consulta Perfil • Eliminar Perfil • Asignar Perfil Usuario 	Colaboradores <ul style="list-style-type: none"> • Usuario

Nota. Autor

Elaborado por Hugo Cajamarca

3.2 Diagrama de componentes

Para una mejor comprensión del producto a implementar se muestra el diagrama de componentes que comprende las dependencias lógicas de software. En este caso está formado por cuatro componentes de software que se interrelacionan entre sí para dar la apariencia de un único sistema ver en la Figura 3.

Diagrama de componentes

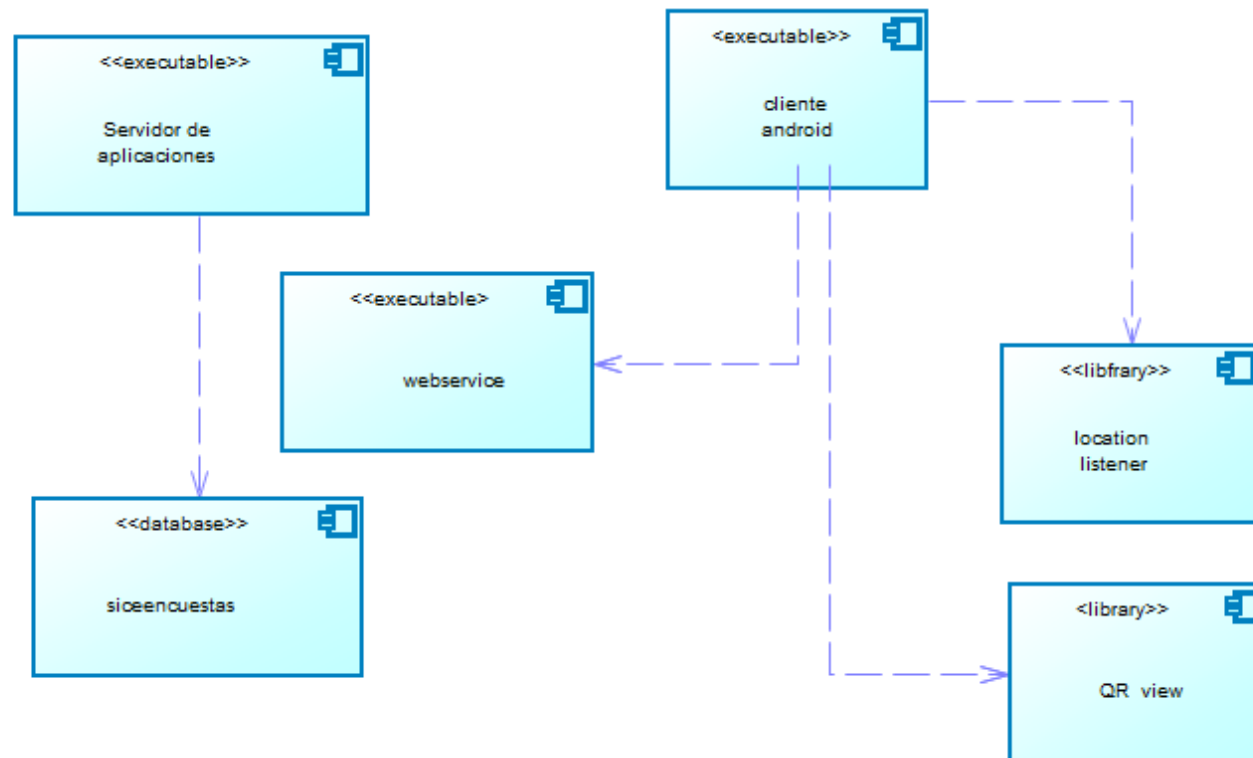


Figura 3. Diagrama de componentes.

Elaborado por Hugo Cajamarca

3.3 Diagramas de casos de uso

Para modelar los casos de uso en el visualizador de obras Salesianas se dividió en dos fases los diagramas de casos de uso del Administrador Visualizador obras Salesianas y los diagramas de casos de uso del cliente del Visualizador obras Salesianas Android se divide para especificar los casos de uso tanto del cliente como del servidor.

Descripción de actores del sistema Administrador Visualizador obras Salesianas

En los siguientes casos de uso, descritos en la tablas 5, se detalla los distintos niveles y tareas en el sistema para realizar la gestión de información en el portal visualizador de obras Salesianas.

Tabla 5.

Casos de uso roles y tareas

Roles	Tareas
Administrador	Usuario asignado a la administración del sistema. Nota: el administrador podrá tener acceso a todos los módulos del sistema para el ingreso, actualización, visualización o eliminación de información o usuarios.
Invitado	Usuario que podrá acceder a consultas únicamente visualizar los códigos QR asignados a cada casa dentro de él portal.

Nota. QR= (quick response code, «código de respuesta rápida»).

Elaborado por Hugo Cajamarca

En los siguientes casos de uso, descritos en la tablas 6, se detalla como el administrador operará el sistema para realizar la gestión de información en el portal visualizador de obras Salesianas.

Tabla 6.

Casos de uso administrador usuario y perfiles

Nombre de caso de uso	Administración de perfiles
Camino Principal Usuarios Administrador	<p>Ingresar al sistema</p> <p>Ingresar usuario y contraseña</p> <p>Visualizar información de las obras salesianas</p> <p>Generar QR.</p> <p>Enviar confirmación de generacion.</p> <p>Editar información.</p> <p>Enviar mensaje de actualización de usuarios.</p> <p>Eliminar información</p> <p>Enviar confirmación de eliminación.</p>
Camino secundario	<p>Datos de usuario incorrectos.</p> <p>Enviar notificación de datos incorrectos.</p> <p>Tipo de datos erróneos.</p> <p>Enviar notificación de tipo de datos erróneos.</p> <p>Enviar notificación de error de generacion.</p> <p>Enviar notificación de error de actualización de usuario.</p> <p>Enviar notificación de error de eliminación.</p>
Precondiciones	El usuario deber estar registrado en el portal
Postcondiciones	Interacción y validación de información entre portal y BDD.

Nota. QR= (quick response code, «código de respuesta rápida»), BDD= Base de datos.

Elaborado por Hugo Cajamarca

Caso de uso ingreso al sistema

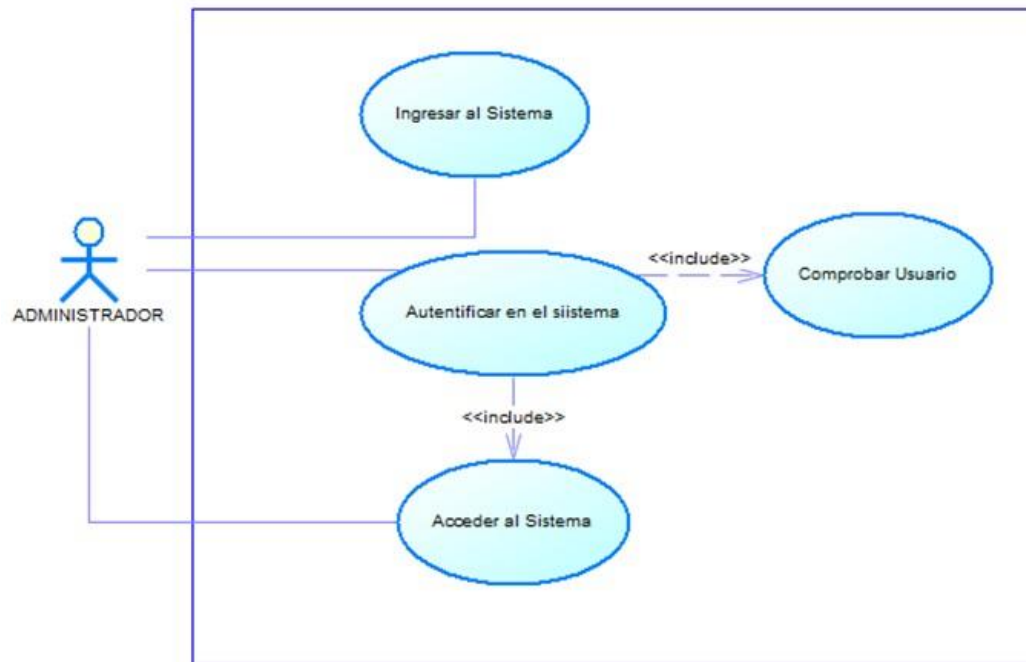


Figura 4. Caso de uso ingreso al sistema

Elaborado por Hugo Cajamarca

En los siguientes casos de uso, descritos en la figura 4, se detalla como el administrador se autentifica en el sistema para realizar la gestión de información en el portal visualizador de obras Salesianas.

Caso de uso generar QR

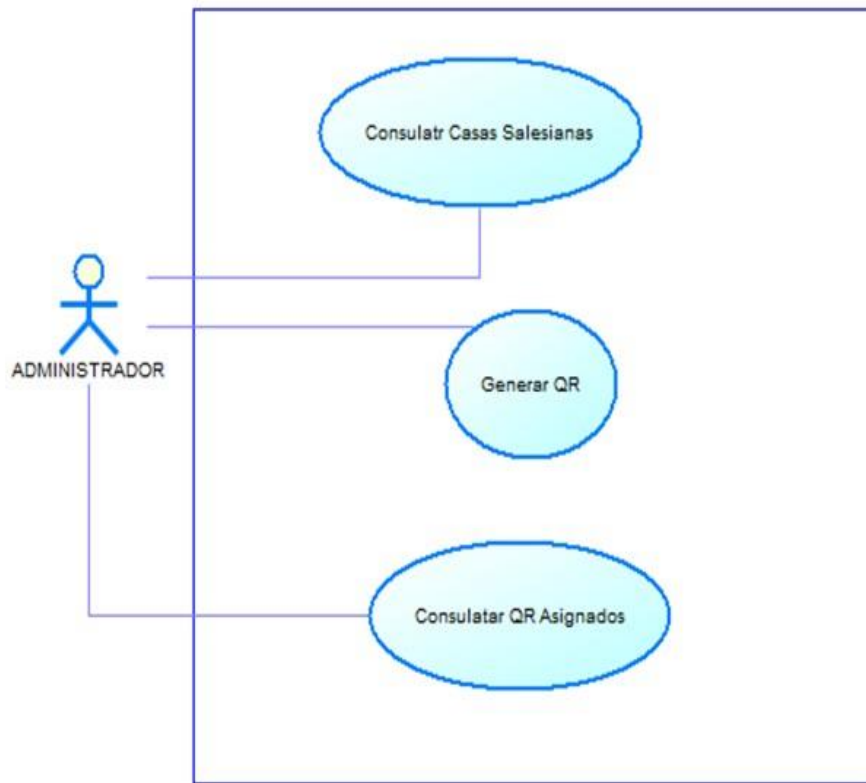
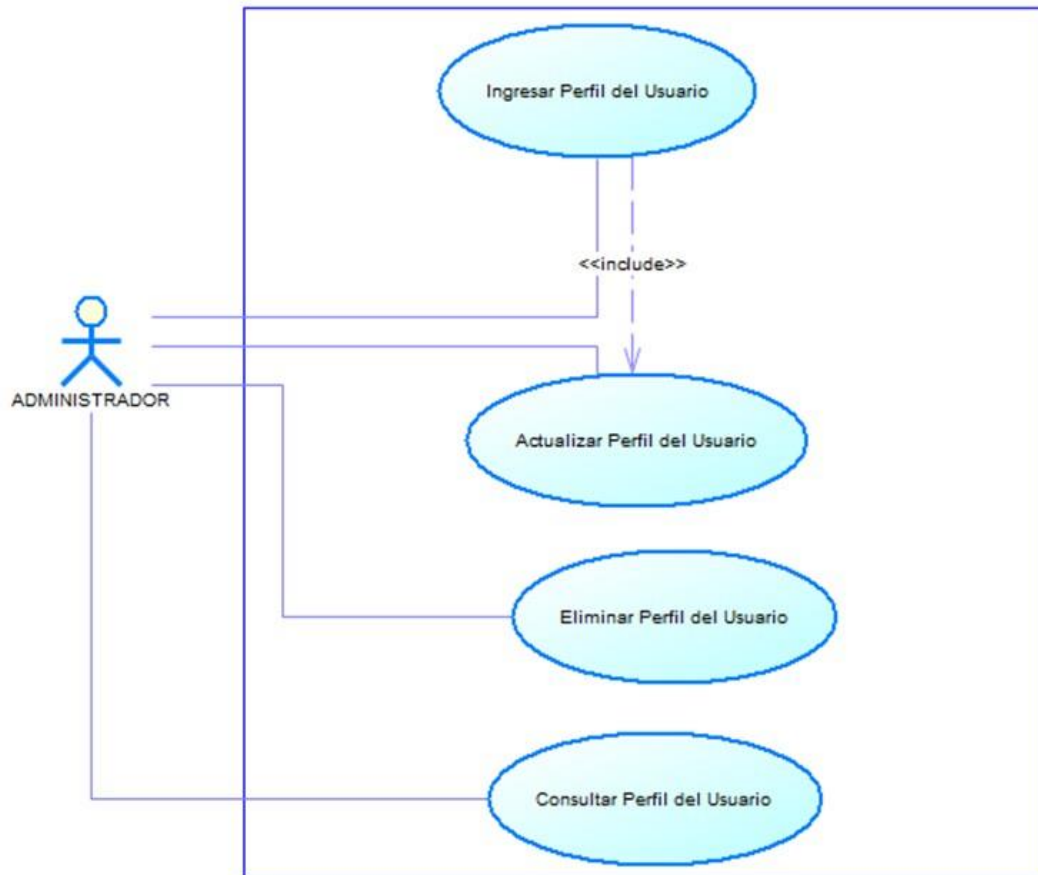


Figura 5. Caso de Uso administrador usuarios y perfiles

Elaborado por Hugo Cajamarca

En los siguientes casos de uso, descritos en la figura 5, se detalla como el administrador opera en el sistema para realizar la gestión de los códigos QR en el portal visualizador de obras Salesianas.

Caso de uso administrado usuarios y perfiles



*Figura 6.*Caso de uso administrado usuarios y perfiles

Elaborado por Hugo Cajamarca

En los siguientes casos de uso, descritos en la figura 6, se detalla como el administrador asigna los perfiles y permisos de navegación en el sistema para realizar la gestión de información en el portal visualizador de obras Salesianas.

Diagrama de casos de uso del cliente Android

Caso de Uso Acceso del cliente al visualizador de obras Salesianas en un Smartphone

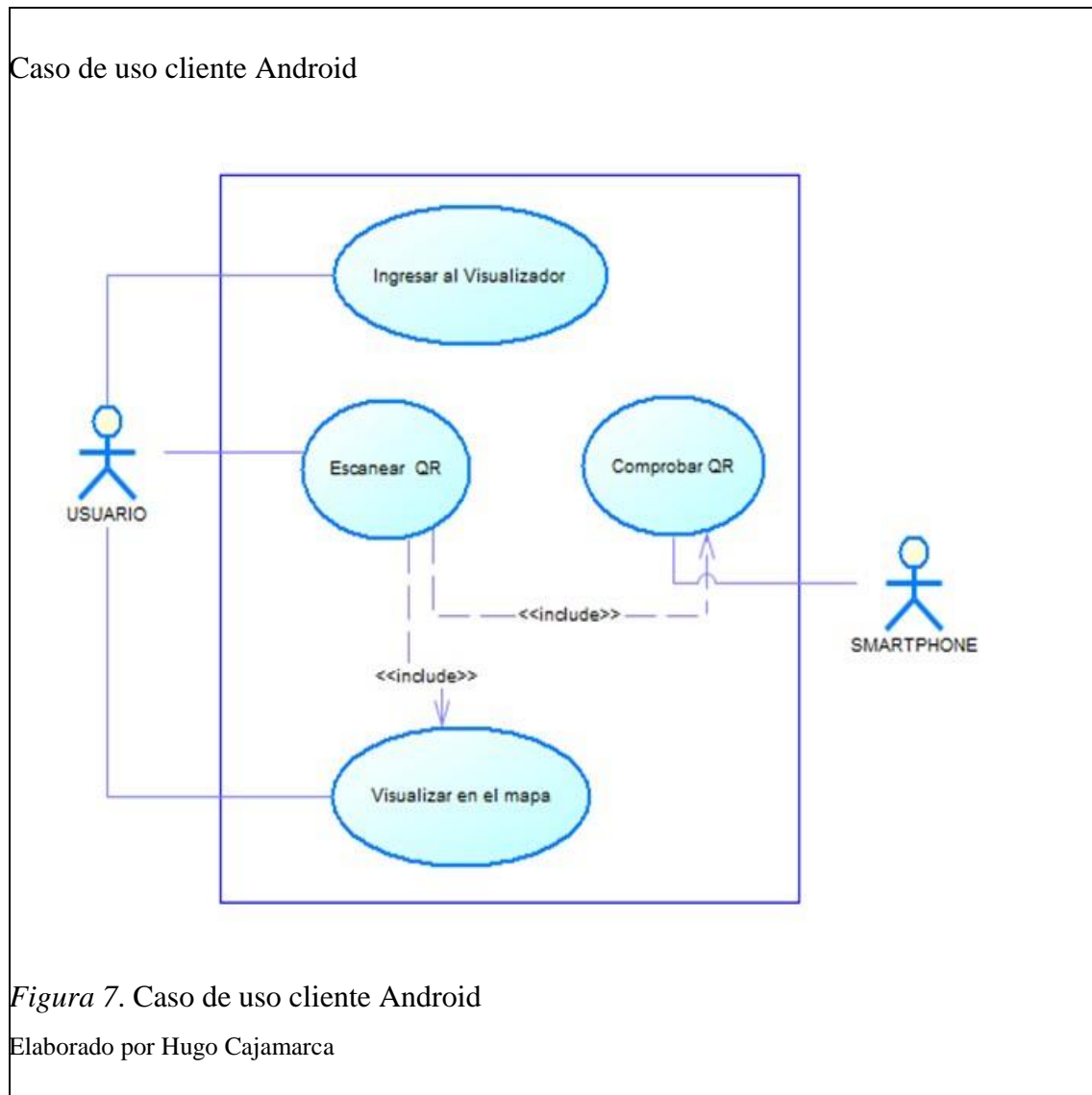


Figura 7. Caso de uso cliente Android

Elaborado por Hugo Cajamarca

En los siguientes casos de uso, descritos en la figura 7, se detalla como el cliente interactúa en el sistema para realizar la gestión de información en el portal visualizador de obras Salesianas a través de su dispositivo móvil.

3.4 Diseño del sistema

3.4.1 Diseño de la base de datos

La base de datos (TesisCar) es la utilizada en el visualizador de obras Salesianas y fue proporcionada por el CIMA-UPS ,solo se utiliza la tabla (tb_lugar) de la cual se le extrae toda la información a mostrar en nuestro visualizador para realizar la integración en nuestro visualizador de obras Salesianas se diseñó una base de datos adicional. La base de datos diseñada tiene como nombre visualizadordb, compuesta de tres tablas. El sistema está codificado en el formato 8-bit Unicode Transformation Format (UTF-8) para no tener problemas con el idioma español. Nos permite manejar las seguridades y gestión de perfiles de los usuarios en el portal y el almacenamiento de los códigos QR generados, en la tabla código QR se efectúa la integración en esta tabla se graba la información extraída de la primera base de datos para posteriormente mostrarle en nuestro aplicativo móvil mediante el escaneo del código QR. El diseño conceptual es una descripción de alto nivel de la estructura de la base de datos independientemente del sistema gestor de base de datos.

Diseño Conceptual de la Base de Datos

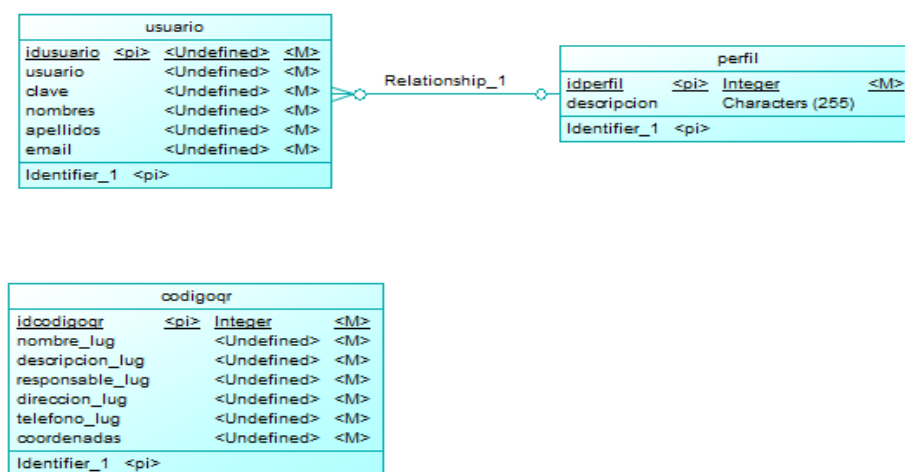
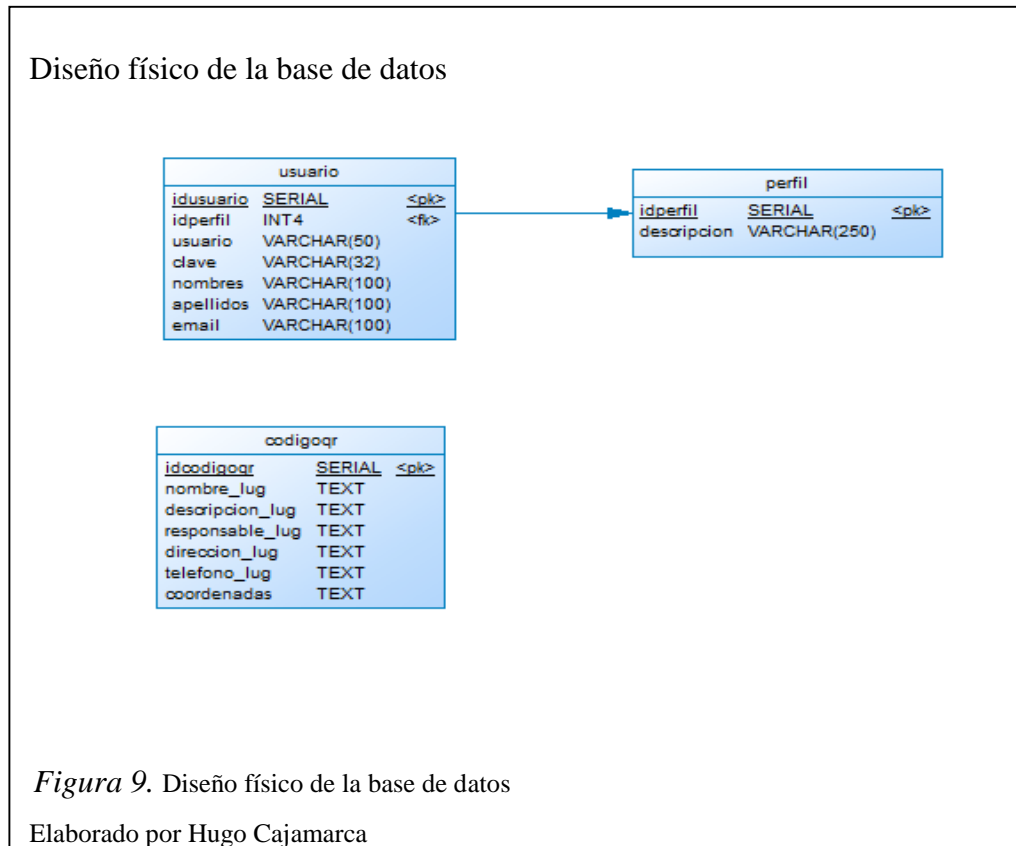


Figura 8.Diseño conceptual

Elaborado por Hugo Cajamarca

Diseño físico de la base de datos para administrar usuarios, perfiles y códigos QR generados.

El modelo físico traduce el diseño conceptual de la base de datos implementado en el sistema gestor de base de datos escogido.



En la figura 10, se detalla la tabla usada de la base de datos proporcionada por el cima y la información a ser utilizada para la gestión de información en el portal visualizador de obras Salesianas y a través de su dispositivo móvil.

Tabla lugar de la base de datos proporcionada por el CIMA

Edit Data - PostgreSQL 9.3 (localhost:5432) - TesisCar - tb_lugar									
	id_lug [PK] serial	id_obr integer	id_elug integer	nombre_lug text	descripcion_lug text	responsable_lug text	direccion_lug text	telefono_lug text	coordenada_lug geometry
1	1	1	7	FUNDACION TAINATE HUASI	Capacitación Ocupacional	OLGER BACA	Av. Natalia Jarrin N	022361234	0104000020E6100000
2	2	2	2	UNIDAD EDUCATIVA MARIO RIZZINI	Educación primaria, bás	BYRON CAMPOVERDE	Av. Natalia Jarrin N	022361234	0104000020E6100000
3	3	3	11	ORATORIO MIGUEL MAGONE	Catequesis	IGNACIO ARRIETA	Av. Natalia Jarrin N	022360064	0104000020E6100000
4	4	4	4	FUNDACIÓN RADIO MENSAJE (1590 kHz AM)	Comunicación Popular	BYRON CAMPOVERDE	Av. Natalia Jarrin N	022360472	0104000020E6100000
5	5	5	5	FUNDACIÓN CASA CAMPESINA CAYAMBE	Promoción Humana	PADRE MARCELO C	Av. Natalia Jarrin N	022361234	0104000020E6100000
6	6	6	6	PARROQUIA SAN JOSÉ DE AYORA	Atención Sacramental: B	HERMANAS DE LA I	Parroquia Ayora (CAY	022360245	0104000020E6100000
7	7	7	6	PARROQUIA SAN JUAN BAUTISTA DE OLMEDO	Atención Sacramental: B	PADRE FRANCISCO	Parroquia Olmedo (CA	022115215	0104000020E6100000
8	8	8	3	CLINICA-MATERNIDAD MITAD DEL MUNDO	Salud médica	CARMEN CADENA	Av. Natalia Jarrin N	022361287	0104000020E6100000
9	9	9	2	ORATORIO MARÍA AUXILIADORA (OMA)	Educación primaria, bás	MG. MÓNICA FONS	Av. Natalia Jarrin N	022361490	0104000020E6100000
10	10	10	8	CENTRO DE APOYO CAYAMBE (UPS)	Centro de Investigación	JANSS BELTRAN	Av. Natalia Jarrin N	''	0104000020E6100000
11	11	11	2	UNIDAD EDUCATIVA SANCHEZ Y CIFUENTES	Educación primaria, bás	RECTORA: MIREYA	Sucre 12-52 y Obispo	062955633	0104000020E6100000
12	12	12	2	CENTRO ARTESANAL MARÍA AUXILIADORA	Ubicado en el barrio Ma	SRA. PIEDAD JAR	Calle Padre Aguirre	072829474	0104000020E6100000
13	13	17	3	CENTRO DE SALUD ADS	Ubicado en el barrio Ma	SRA. MARLENE FI	Calle Padre Aguirre	072841007	0104000020E6100000
14	14	16	2	ESCUELA CARLOS CRESPI	Ubicado en el barrio Ma	LIC. WILSON MOS	Tarqui y Rafael Mari	072842692	0104000020E6100000
15	15	13	6	PARROQUIA MARÍA AUXILIADORA	Ubicado en el barrio Ma	PADRE RAÚL CONZ	Calle Padre Aguirre	072823298	0104000020E6100000
16	16	14	6	CASA DE LOS MISIONEROS ENFERMOS	Ubicado en el barrio Ma	PADRE AMBROSSIO	Muñoz Vernaza 13-33	072823812	0104000020E6100000
17	17	15	6	CASA DE LA JUVENTUD	Ubicado en Paute	LIC. SANDRO GAV	Paute	072250684	0104000020E6100000
18	18	18	2	COLEGIO A DISTANCIA JUAN LUNARDI	Ubicado camino a Paute	LIC. OSWALDO FUJ	Via a Paute Uzhupud	072228046	0104000020E6100000
19	19	19	2	COLEGIO AGRONÓMICO SALESIANO	Ubicado camino a Paute	DR. SEGUNDO TOL	Via a Paute Uzhupud	072228350	0104000020E6100000
20	20	20	9	EDITORIAL DON BOSCO	Ubicado en el barrio Ma		Calle Padre Aguirre	0	0104000020E6100000
21	21	21	2	COLEGIO A DISTANCIA MARIO RIZZINI	Ubicado en el barrio Ya		Av. Don Bosco y Feli	072888141	0104000020E6100000
22	22	22	7	FUNDACIÓN SALESIANA FACES	Ubicada junto a la UPS	ARQ. EDGAR GOR	Calle Vieja y Elia L	072862213	0104000020E6100000

Figura 10. Base de datos proporcionada por el CIMA

Elaborado por Hugo Cajamarca

El sistema gestor de base de datos seleccionado es Postgresql porque es de libre distribución y multiplataforma es una de las más potentes que existe, soporta transacciones, vistas, triggers, joins, foreign keys y stored procedures. PostgreSQL incluye un modelo de seguridad completo ya que soporta un protocolo de comunicación encriptado por SSL.

CAPÍTULO 4

DESARROLLO

4.1 Desarrollo del módulo de administración

En esta etapa se detalla el desarrollo del módulo administrador del visualizador de obras Salesianas, la parte fundamental de la aplicación es el manejo de sesiones, perfiles y la generación de los códigos QR.

4.1.1 Funcionalidades del módulo administrador

- Administrar sesiones, pues de esta manera se controla el acceso al visualizador de obras Salesianas.
- Administrar usuarios, roles, perfiles y contraseñas.
- Administrar los QR.

4.1.2 Acceso al sistema

Para controlar el acceso al sistema se tiene el uso de variables de sesión, en la aplicación cuando un usuario quiere ingresar al sistema se autentifica, se comprueba las credenciales que ingresó y si corresponden a un usuario registrado en la base de datos, se crea una variable de sesión que se llama “usuario”, la misma que contiene todos los datos del usuario ingresado, esto se puede ver en la figura 10.

Clase autenticación

Figura 11. Clase autenticación

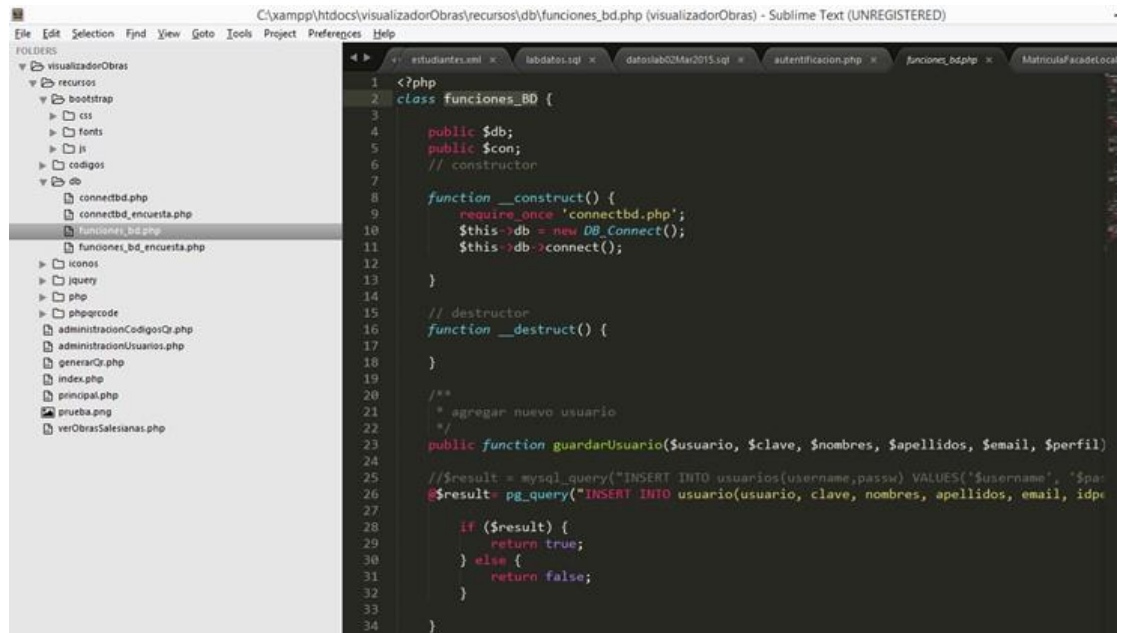
Elaborado por Hugo Cajamarca

En la figura 11 se puede observar el método empleado. En la línea 16 se consultan todos los datos del usuario actual; en la línea 17 se crea la variable de sesión y se inserta la información consultada en la base de datos, en la línea 18 se re direcciona a la página principal de la aplicación.

Funciones BD

La Clase creada para verificar que un usuario está registrado en el sistema es funciones_BD como se observa en la figura 12.

Clase función BD



```
1 <?php
2 class funciones_bd {
3
4     public $db;
5     public $con;
6     // constructor
7
8     function __construct() {
9         require_once 'connectbd.php';
10        $this->db = new DB_Connect();
11        $this->db->connect();
12    }
13
14    // destructor
15    function __destruct() {
16    }
17
18    /**
19     * agregar nuevo usuario
20     */
21    public function guardarUsuario($usuario, $clave, $nombres, $apellidos, $email, $perfil)
22    {
23        // $result = mysql_query("INSERT INTO usuarios(username,password) VALUES('$usuario', '$password')");
24        $result = pg_query("INSERT INTO usuario(usuario, clave, nombres, apellidos, email, idperfil) VALUES('$usuario', '$clave', '$nombres', '$apellidos', '$email', '$perfil')");
25        if ($result) {
26            return true;
27        } else {
28            return false;
29        }
30    }
31
32 }
33
34 }
```

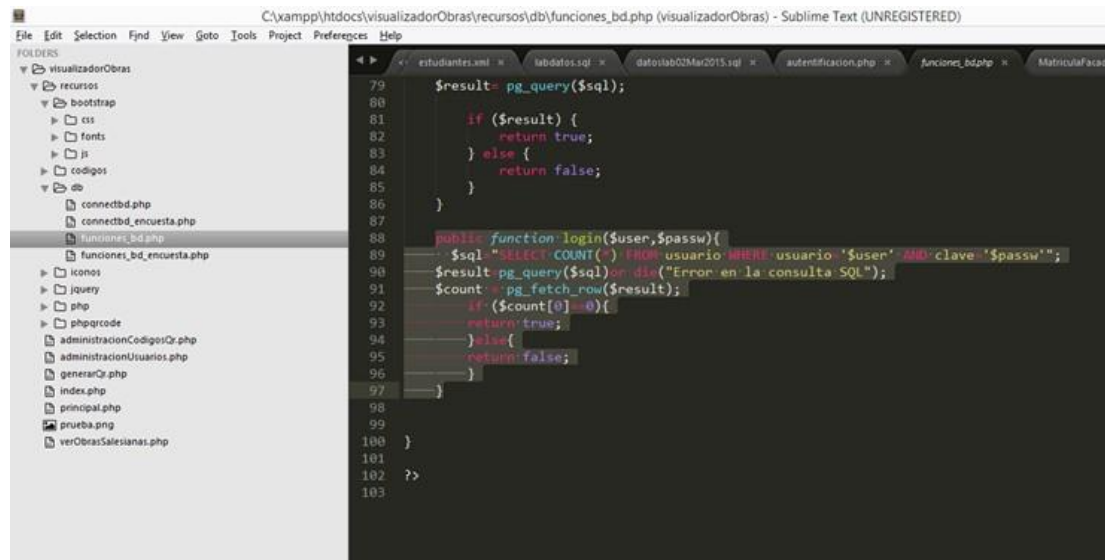
Figura 12. Clase función BD

Elaborado por Hugo Cajamarca

En la figura 12 se puede observar que contiene una serie de métodos, los mismos que sirven para realizar transacciones en la base de datos, esta clase es invocada de varios archivos .php los mismos que reutilizan los métodos que esta clase contiene.

Busca coincidencias entre el usuario y la clave ingresadas en el sistema y devuelve un True o False según sea el caso como se observa en la figura 13.

Clase función BD



```
79 $result= pg_query($sql);
80
81 if ($result) {
82     return true;
83 } else {
84     return false;
85 }
86 }
87
88 public function login($user,$passw){
89     $sql="SELECT COUNT(*) FROM usuario WHERE usuario='$user' AND clave='$passw'";
90     $result= pg_query($sql) or die("Error en la consulta SQL");
91     $count= pg_fetch_row($result);
92     if ($count[0]!=0){
93         return true;
94     }else{
95         return false;
96     }
97 }
98
99
100 }
101
102 ?>
103
```

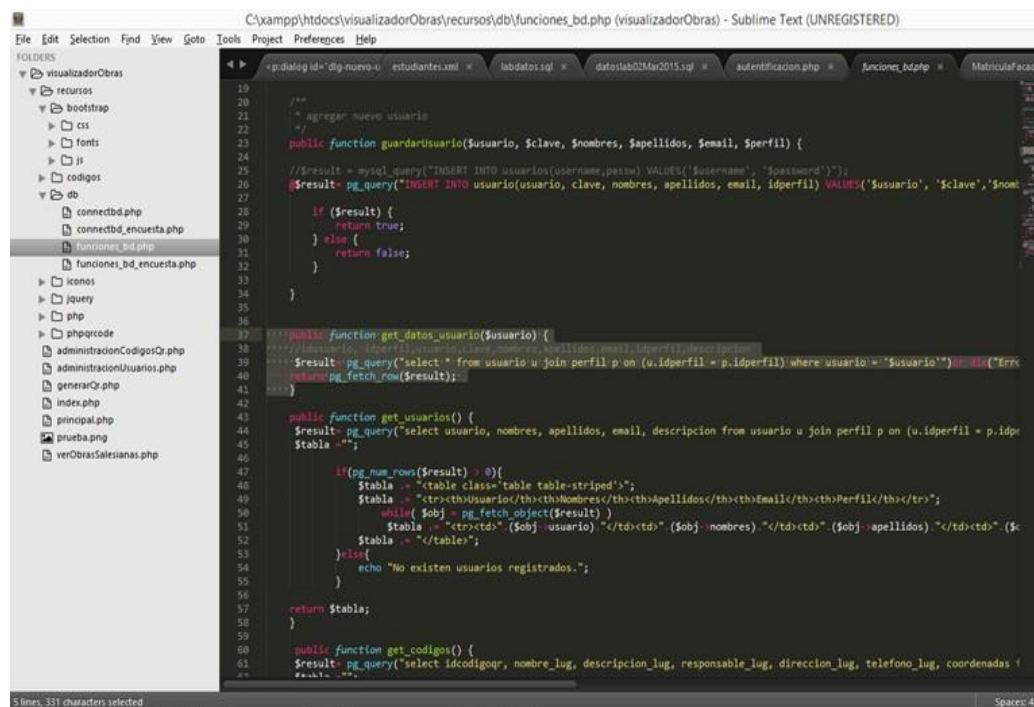
Figura 13. Clase Función BD

Elaborado por Hugo Cajamarca

En la figura 13 se puede observar El método public function login que se encarga de hacer esta verificación.

El método public function get_datos_usuario gestiona las credenciales del usuario desde la base de datos como se puede observar en la figura 14.

Clase función BD



```
19
20 /**
21  * agregar nuevo usuario
22  */
23 public function guardarUsuario($usuario, $clave, $nombres, $apellidos, $email, $perfil) {
24
25     // $result = mysql_query("INSERT INTO usuarios(username,password) VALUES('$usuario', '$password')");
26     $result = pg_query("INSERT INTO usuario(usuario, clave, nombres, apellidos, email, idperfil) VALUES('$usuario', '$clave', '$nombres', '$apellidos', '$email', '$idperfil')");
27
28     if ($result) {
29         return true;
30     } else {
31         return false;
32     }
33 }
34
35
36
37 public function get_datos_usuario($usuario) {
38     // $result = mysql_query("select * from usuarios where username = '$usuario'");
39     $result = pg_query("select * from usuario u join perfil p on (u.idperfil = p.idperfil) where usuario = '$usuario'");
40     return pg_fetch_row($result);
41 }
42
43
44 public function get_usuarios() {
45     $result = pg_query("select usuario, nombres, apellidos, email, descripcion from usuario u join perfil p on (u.idperfil = p.idperfil)");
46     $tabla = "";
47
48     if (pg_num_rows($result) > 0) {
49         $tabla = "<table class='table table-striped'>";
50         $tabla = "<tr><th>usuario</th><th>nombres</th><th>apellidos</th><th>email</th><th>perfil</th></tr>";
51         while ($obj = pg_fetch_object($result)) {
52             $tabla = "<tr><td>{$obj->usuario}</td><td>{$obj->nombres}</td><td>{$obj->apellidos}</td><td>{$obj->email}</td><td>{$obj->perfil}</td></tr>";
53         }
54         $tabla = "</table>";
55     } else {
56         echo "No existen usuarios registrados.";
57     }
58
59     return $tabla;
60 }
61
62 public function get_codigos() {
63     $result = pg_query("select idcodigoqr, nombre_lug, descripcion_lug, responsable_lug, direccion_lug, telefono_lug, coordenadas from codigos");
64 }
```

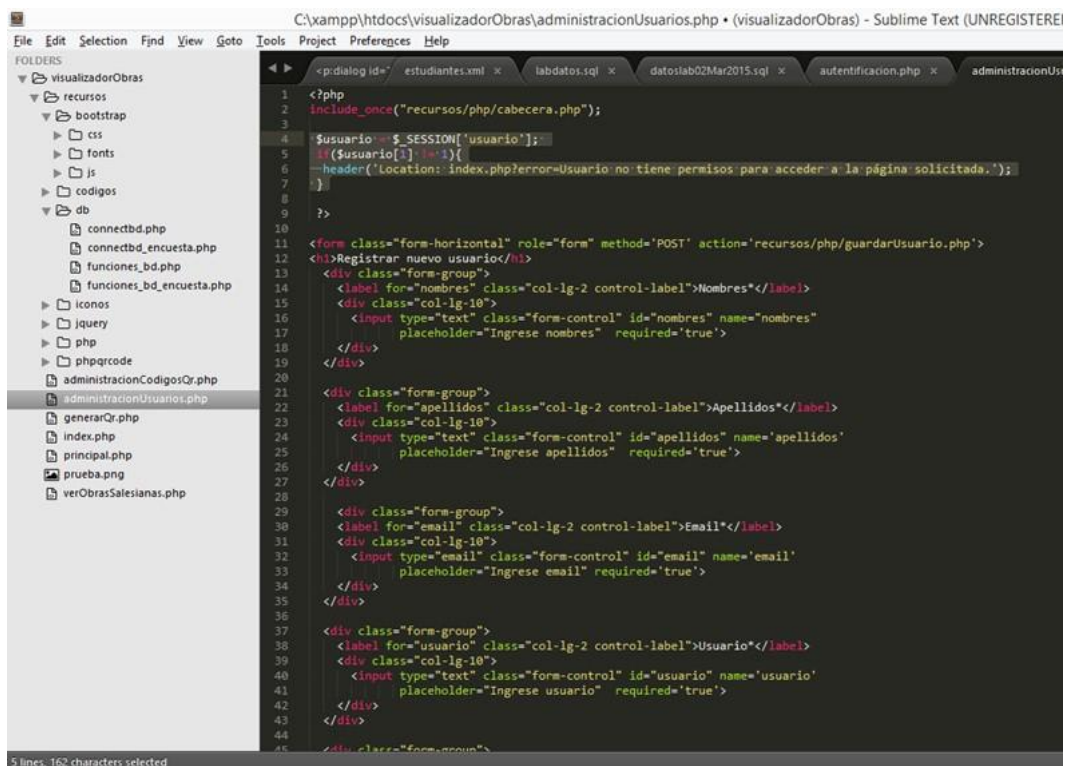
Figura 14. Clase función BD

Fuente Autor

Elaborado por Hugo Cajamarca

La clase class funciones_BD consulta todos los datos del usuario actual en la misma se incluye el perfil del usuario, el mismo que autoriza o restringe el acceso a los módulos del sistema, por ejemplo Si alguien quiere acceder directamente a una url del sistema pero su perfil no se lo permite este es re direccionado a la página principal del sistema como se visualiza en la figura 15.

Clase administración usuarios



The image shows a Sublime Text editor window with the title bar 'C:\xampp\htdocs\visualizadorObras\administracionUsuarios.php • (visualizadorObras) - Sublime Text (UNREGISTERED)'. The left sidebar displays a file explorer for the 'visualizadorObras' directory, listing folders like 'recursos', 'bootstrap', 'css', 'fonts', 'js', 'codigos', 'db', and various PHP files including 'connectbd.php', 'funciones_bd.php', 'administracionUsuarios.php', and others. The main editor area shows the code for 'administracionUsuarios.php'. The code starts with a PHP header, includes 'recursos/php/cabecera.php', and checks if the user is logged in. If not, it displays an error message. Otherwise, it renders a registration form with fields for 'Nombres', 'Apellidos', 'Email', and 'Usuario'. The form uses Bootstrap classes for styling and includes validation attributes like 'required'.

```
<?php
include_once("recursos/php/cabecera.php");

$usuario = $_SESSION['usuario'];
if ($usuario != null) {
    header('Location: index.php?error=Usuario no tiene permisos para acceder a la página solicitada.');
```

```
<form class="form-horizontal" role="form" method="POST" action="recursos/php/guardarUsuario.php">
<h3>Registrar nuevo usuario</h3>
<div class="form-group">
    <label for="nombres" class="col-lg-2 control-label">Nombres</label>
    <div class="col-lg-10">
        <input type="text" class="form-control" id="nombres" name="nombres"
            placeholder="Ingrese nombres" required="true">
    </div>
</div>

<div class="form-group">
    <label for="apellidos" class="col-lg-2 control-label">Apellidos</label>
    <div class="col-lg-10">
        <input type="text" class="form-control" id="apellidos" name="apellidos"
            placeholder="Ingrese apellidos" required="true">
    </div>
</div>

<div class="form-group">
    <label for="email" class="col-lg-2 control-label">Email</label>
    <div class="col-lg-10">
        <input type="email" class="form-control" id="email" name="email"
            placeholder="Ingrese email" required="true">
    </div>
</div>

<div class="form-group">
    <label for="usuario" class="col-lg-2 control-label">Usuario</label>
    <div class="col-lg-10">
        <input type="text" class="form-control" id="usuario" name="usuario"
            placeholder="Ingrese usuario" required="true">
    </div>
</div>
</div>
```

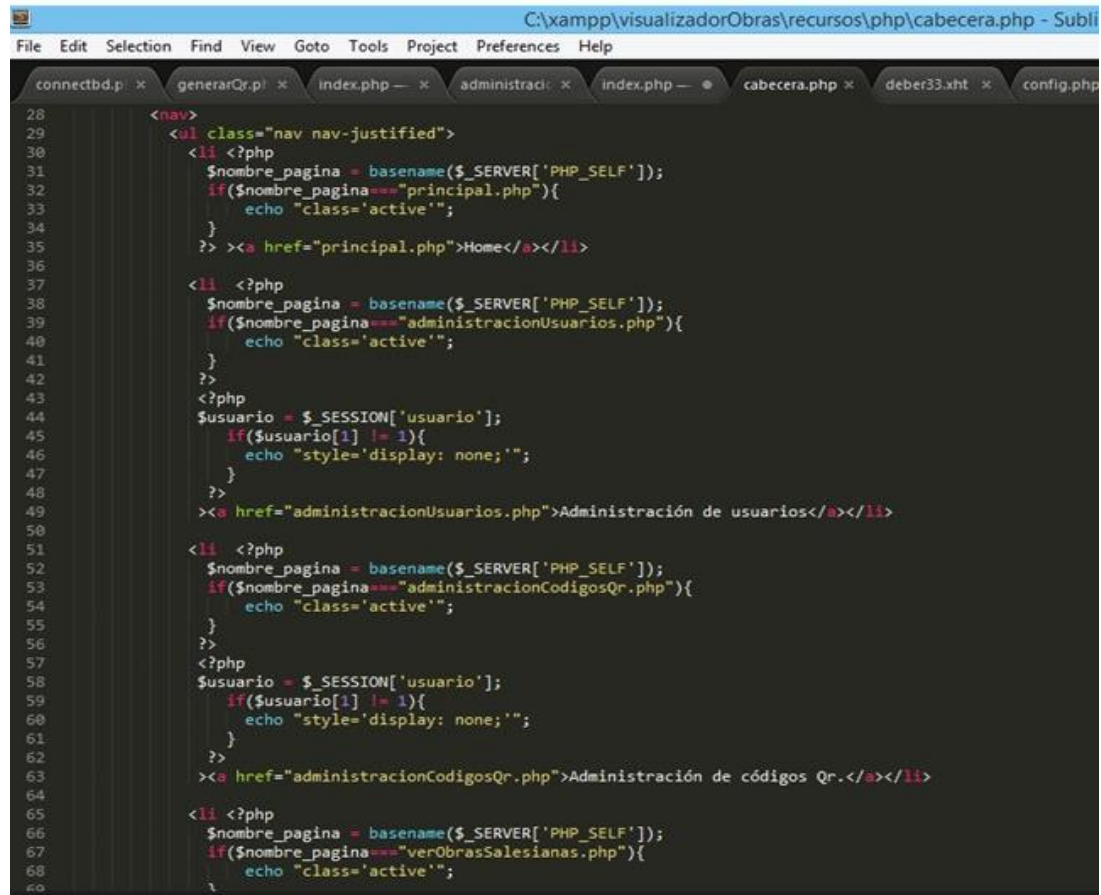
Figura 15. Clase administración usuarios

Elaborado por Hugo Cajamarca

En la figura 15 se puede observar la validación, caso contrario se despliega el mensaje Usuario no tiene permisos para acceder a la página solicitada.

Los permisos de navegación se pueden observar en el archivo cabecera.php, este archivo es reutilizado en cada una de la página con el fin de obtener el mismo estilo en todas la paginas (plantilla) y desplegar el menú al que corresponde cada usuario como se observa en la figura 16.

Clase cabecera



```
28 <nav>
29 <ul class="nav nav-justified">
30 <li <?php
31     $nombre_pagina = basename($_SERVER['PHP_SELF']);
32     if($nombre_pagina=="principal.php"){
33         echo "class='active'";
34     }
35     ?> <a href="principal.php">Home</a></li>
36
37 <li <?php
38     $nombre_pagina = basename($_SERVER['PHP_SELF']);
39     if($nombre_pagina=="administracionUsuarios.php"){
40         echo "class='active'";
41     }
42     ?>
43 <?php
44 $usuario = $_SESSION['usuario'];
45 if($usuario[1] != 1){
46     echo "style='display: none;'";
47 }
48 ?>
49 <a href="administracionUsuarios.php">Administración de usuarios</a></li>
50
51 <li <?php
52     $nombre_pagina = basename($_SERVER['PHP_SELF']);
53     if($nombre_pagina=="administracionCodigosQr.php"){
54         echo "class='active'";
55     }
56     ?>
57 <?php
58 $usuario = $_SESSION['usuario'];
59 if($usuario[1] != 1){
60     echo "style='display: none;'";
61 }
62 ?>
63 <a href="administracionCodigosQr.php">Administración de códigos Qr.</a></li>
64
65 <li <?php
66     $nombre_pagina = basename($_SERVER['PHP_SELF']);
67     if($nombre_pagina=="verObrasSalesianas.php"){
68         echo "class='active'";
69     }
70     ?>
71 <a href="verObrasSalesianas.php">verObrasSalesianas.php</a></li>
72 </ul>
73 </nav>
```

Figura 16. Clase cabecera

Elaborado por Hugo Cajamarca

En la figura 16 se puede observar que se verifica el perfil del usuario actual para desplegar el menú, caso contrario no lo muestra, lo oculta mediante css exactamente en la línea 45 y 59.

Para verificar la validez de la autorización en el módulo administrador integrado, se definieron dos tipos de perfiles que son perfil administrador y perfil invitado.

Ingreso al sistema con el perfil administrador

Página principal perfil administrador



Figura 17. Página principal perfil administrador

Elaborado por Hugo Cajamarca

En la figura 17 se puede observar la página principal con el menú del usuario administrador que le permite navegar en el sistema con el perfil asignado

Ingreso al sistema con el perfil invitado

Página principal perfil invitado



Figura 18. Página principal perfil invitado

Elaborado por Hugo Cajamarca

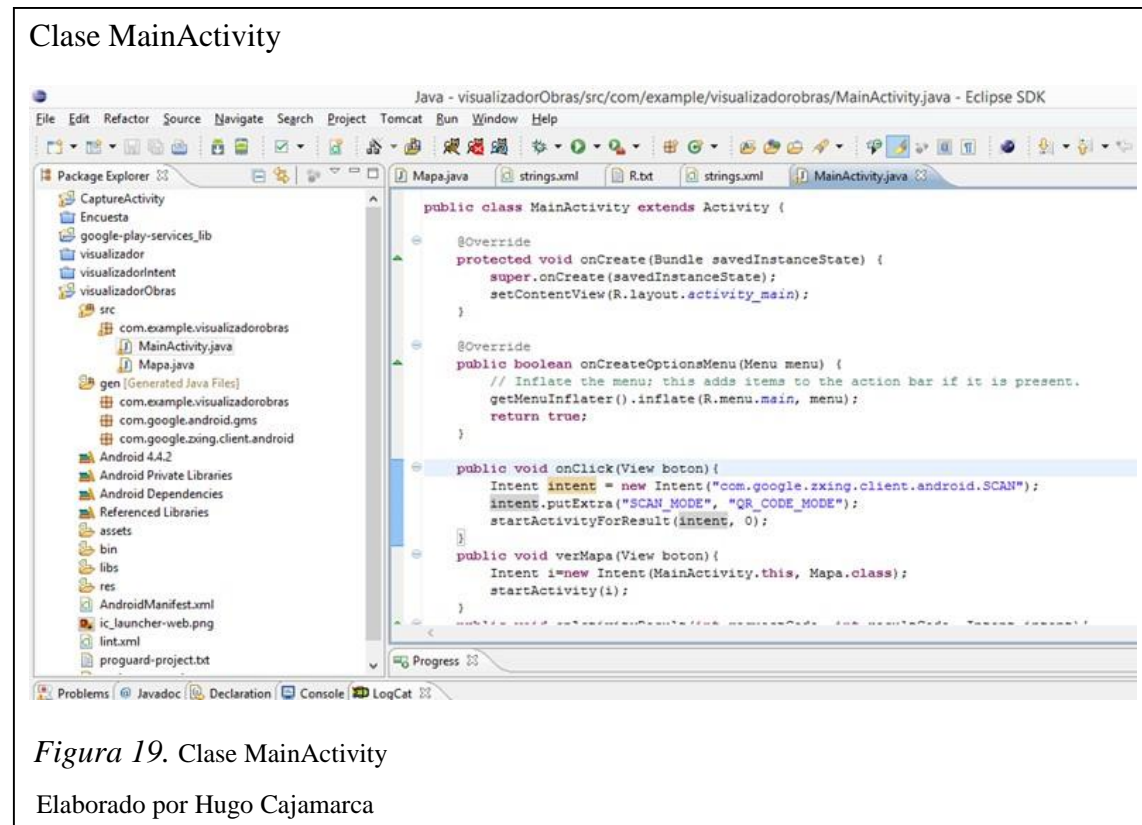
En la figura 18 se puede observar la página principal con el menú del usuario invitado que le permite navegar en el sistema con el perfil asignado

4.2 Desarrollo módulo cliente

Durante esta etapa se desarrolló el sistema cliente del visualizador de obras Salesianas, La parte fundamental de la aplicación móvil es la integración de la librería zxing a un proyecto además de la librería google maps v2.0 solo se detallara las partes más importantes del código fuente.

4.2.1 Clase MainActivity

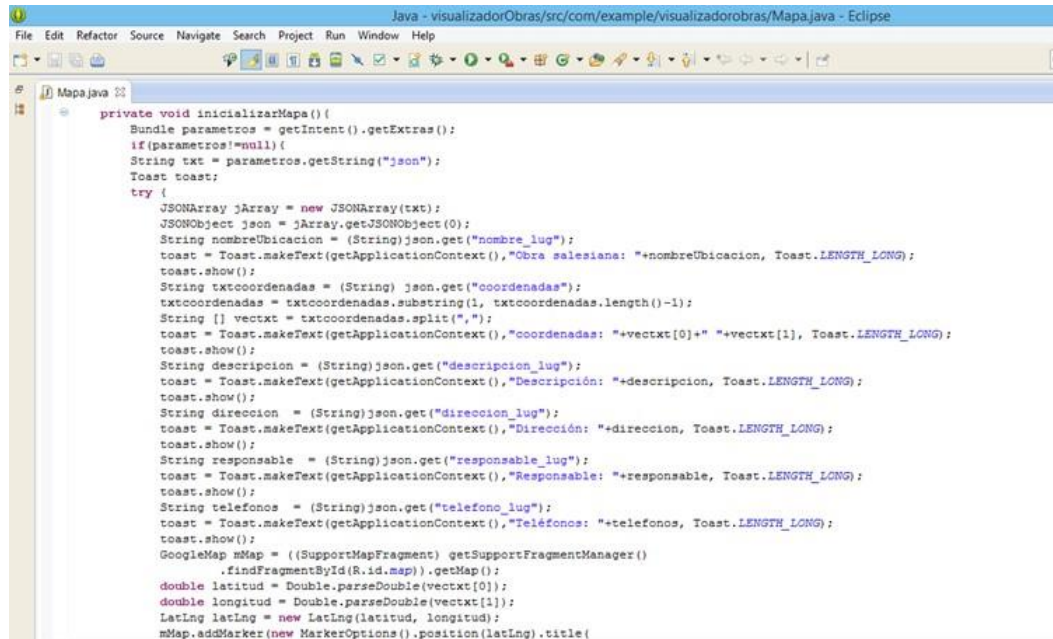
Una vez integrada las 2 librerías a la aplicación es utilizarlas y eso lo hacemos en la clase MainActivity.java



4.2.2 Clase Mapa

Para establecer la visualización de la información en el mapa a través del escaneo del código QR se implementa el método “inicializarMapa” que permite mostrar la información en el dispositivo móvil.

Clase Mapa



```
private void inicializarMapa(){
    Bundle parametros = getIntent().getExtras();
    if(parametros!=null){
        String txt = parametros.getString("json");
        Toast toast;
        try {
            JSONArray jArray = new JSONArray(txt);
            JSONObject json = jArray.getJSONObject(0);
            String nombreUbicacion = (String)json.get("nombre_lug");
            toast = Toast.makeText(getApplicationContext(),"Obra salesiana: "+nombreUbicacion, Toast.LENGTH_LONG);
            toast.show();
            String txtcoordenadas = (String) json.get("coordenadas");
            txtcoordenadas = txtcoordenadas.substring(1, txtcoordenadas.length()-1);
            String [] vectxt = txtcoordenadas.split(",");
            toast = Toast.makeText(getApplicationContext(),"coordenadas: "+vectxt[0]+" "+vectxt[1], Toast.LENGTH_LONG);
            toast.show();
            String descripcion = (String)json.get("descripcion_lug");
            toast = Toast.makeText(getApplicationContext(),"Descripción: "+descripcion, Toast.LENGTH_LONG);
            toast.show();
            String direccion = (String)json.get("direccion_lug");
            toast = Toast.makeText(getApplicationContext(),"Dirección: "+direccion, Toast.LENGTH_LONG);
            toast.show();
            String responsable = (String)json.get("responsable_lug");
            toast = Toast.makeText(getApplicationContext(),"Responsable: "+responsable, Toast.LENGTH_LONG);
            toast.show();
            String telefonos = (String)json.get("telefono_lug");
            toast = Toast.makeText(getApplicationContext(),"Teléfonos: "+telefonos, Toast.LENGTH_LONG);
            toast.show();
            GoogleMap mMap = ((SupportMapFragment) getSupportFragmentManager()
                .findFragmentById(R.id.msp)).getMap();
            double latitud = Double.parseDouble(vectxt[0]);
            double longitud = Double.parseDouble(vectxt[1]);
            LatLng latLng = new LatLng(latitud, longitud);
            mMap.addMarker(new MarkerOptions().position(latLng).title{
```

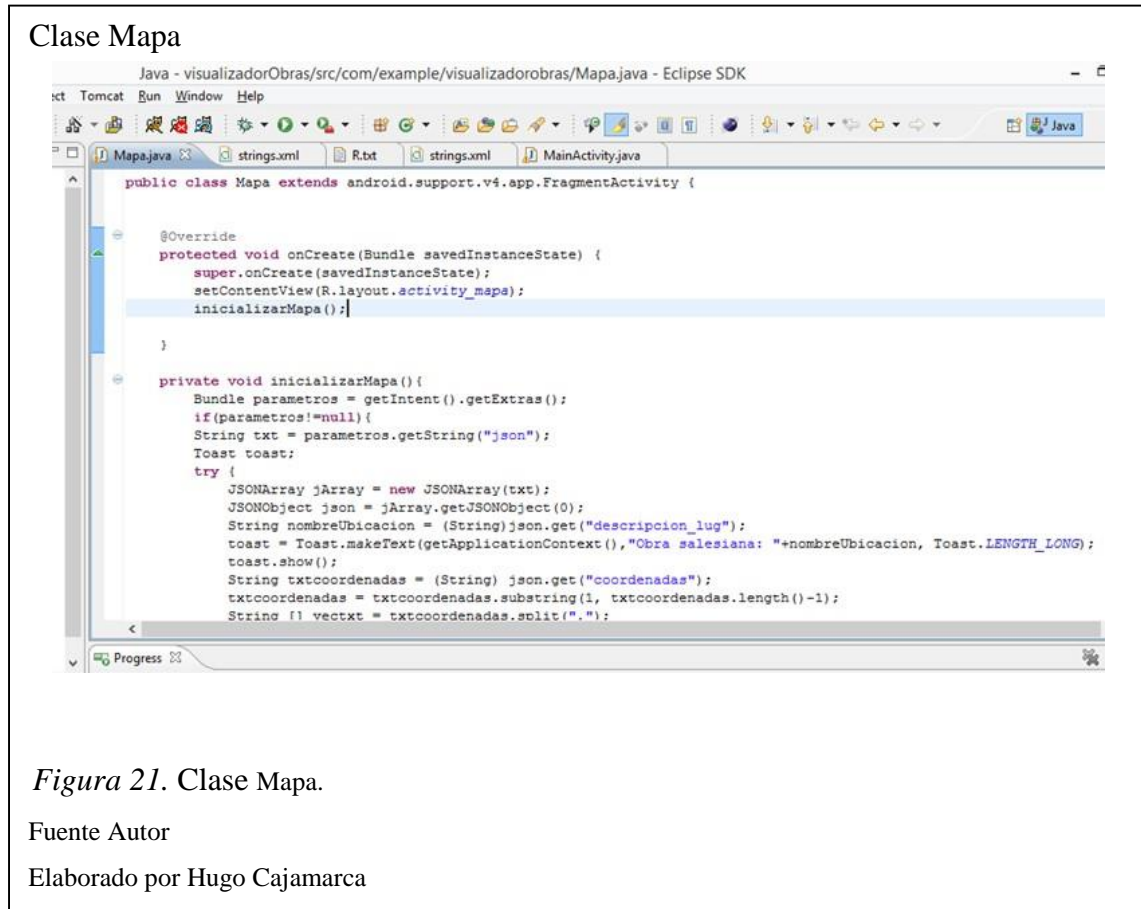
Figura 20. Clase Mapa

Elaborado por Hugo Cajamarca

En la figura 20 se puede observar la acción que llama la librería .google .zxing .client Android.

4.2.3 Clase Mapa

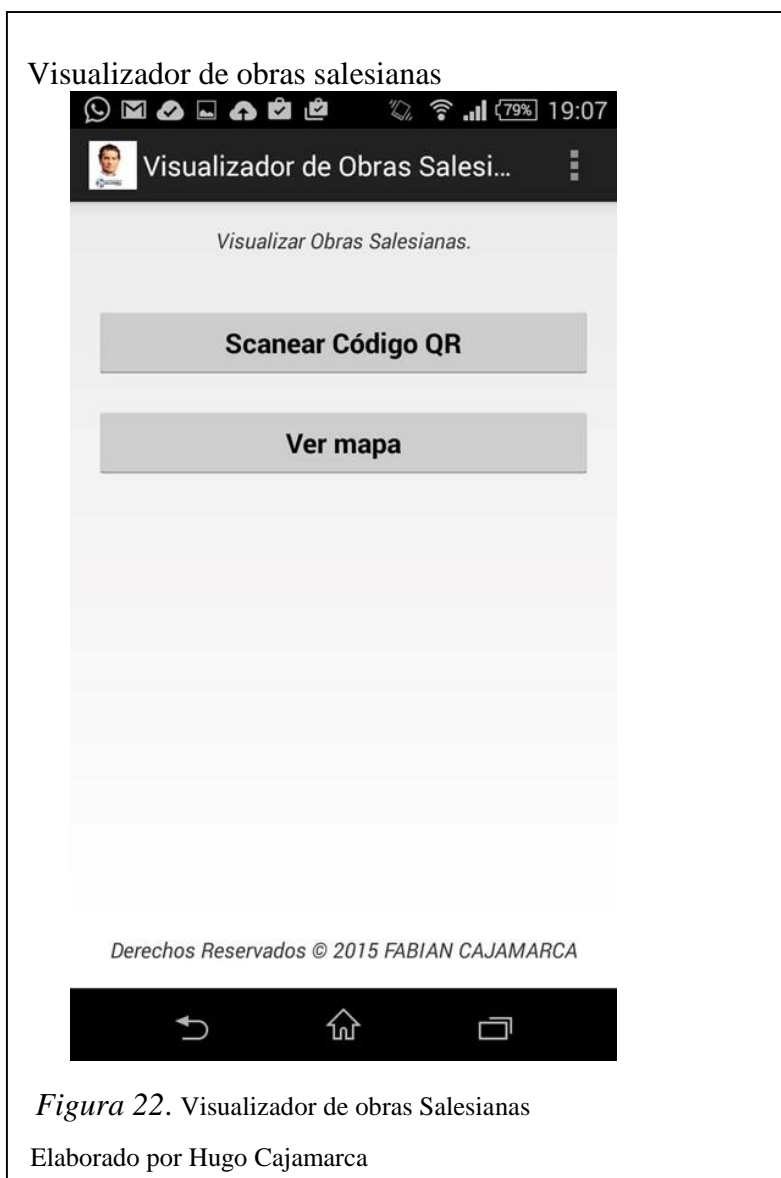
La actividad mapa.java recibe el parámetro enviado y inicializa el mapa colocando las coordenadas enviadas.



En la figura 21 podemos ver el método onCreate que llama al método inicializarMapa();

4.2.4 Visualizador de obras Salesianas

Para poder escanear los códigos QR generados se desarrolló un visualizador para Android



Menús y submenús a los que tiene acceso el usuario administrador



Figura 23. Menús y submenús

Elaborado por Hugo Cajamarca

En la figura 23, se detalla menús y submenús que tiene el perfil administrador

Agregar nuevo usuario permite asignar los permisos, modificar, eliminar y visualizar que tendrá cada uno de los usuarios a los menús y a los submenús.

Menú administración

The screenshot shows a web application interface for user registration. At the top, there is a navigation bar with the following menu items: Home, Administración de usuarios, Administración de códigos QR, Obras Salesianas, and a user profile dropdown for 'Administrador: HUGO FABIAN'. The 'Registrar nuevo usuario' form contains several input fields: 'Apellidos*', 'Email*', 'Usuario*', 'Clave*', and a 'Perfil*' dropdown menu. The 'Perfil*' dropdown is currently open, showing two options: 'Administrador' and 'Invitado'. Below the form is a table displaying a list of existing users. Two green boxes with arrows highlight specific parts of the interface: one box around the top navigation bar points to the text 'Menú de administración', and another box around the 'Perfil*' dropdown points to the text 'Perfil: puede elegir administrador o invitado'.

Usuario	Nombres	Apellidos	Email	Perfil
fcajamarca	FABIAN	CAJAMARCA	cajamarca86@hotmail.com	Invitado
hcajamarca	HUGO FABIAN	CAJAMARCA QUSHPE	hcajamarca@bce.ec	Administrador

Figura 24. Menú administración

Elaborado por Hugo Cajamarca

En la figura 24, se detalla el menú administración, submenú administración de usuarios

Menú administración, submenú administración códigos QR

Agregar nuevo QR permite elegir una casa salesiana asignar el QR, modificar, eliminar y visualizar que tendrá cada uno de las casas Salesianas.

Menú administración, submenú obras Salesianas

The screenshot shows the 'Administración de códigos QR' menu. A green box highlights the 'Generar código Qr' button. Another green box highlights the 'Lista de obras' dropdown menu, which is open, showing a list of Salesian works. A green arrow points from the 'Generar código Qr' button to a text box on the right that says 'Menú de administración' and 'Submenú Administrar códigos QR'. Another green arrow points from the 'Lista de obras' dropdown to a text box on the right that says 'Asignar códigos QR'.

Menú de administración

Submenú Administrar códigos QR

Asignar códigos QR

Figura 25. Menú administración, submenú obras Salesianas

Elaborado por Hugo Cajamarca

En la figura 25, se detalla el menú administración, submenú administración de códigos QR.

Lista casa Salesianas con su QR asignado permite visualizar el código QR asignado a cada casa Salesiana.

Menú administración, submenú obras Salesianas

The screenshot shows the 'Lista de obras salesianas' page. A green box highlights the 'Lista de obras salesianas' title. Another green box highlights the 'Descargar el QR' button. A green arrow points from the 'Lista de obras salesianas' title to a text box on the right that says 'Menú administrador submenú obras salesianas'. Another green arrow points from the 'Descargar el QR' button to a text box on the right that says 'Descargar el QR'.

Menú administrador submenú obras salesianas

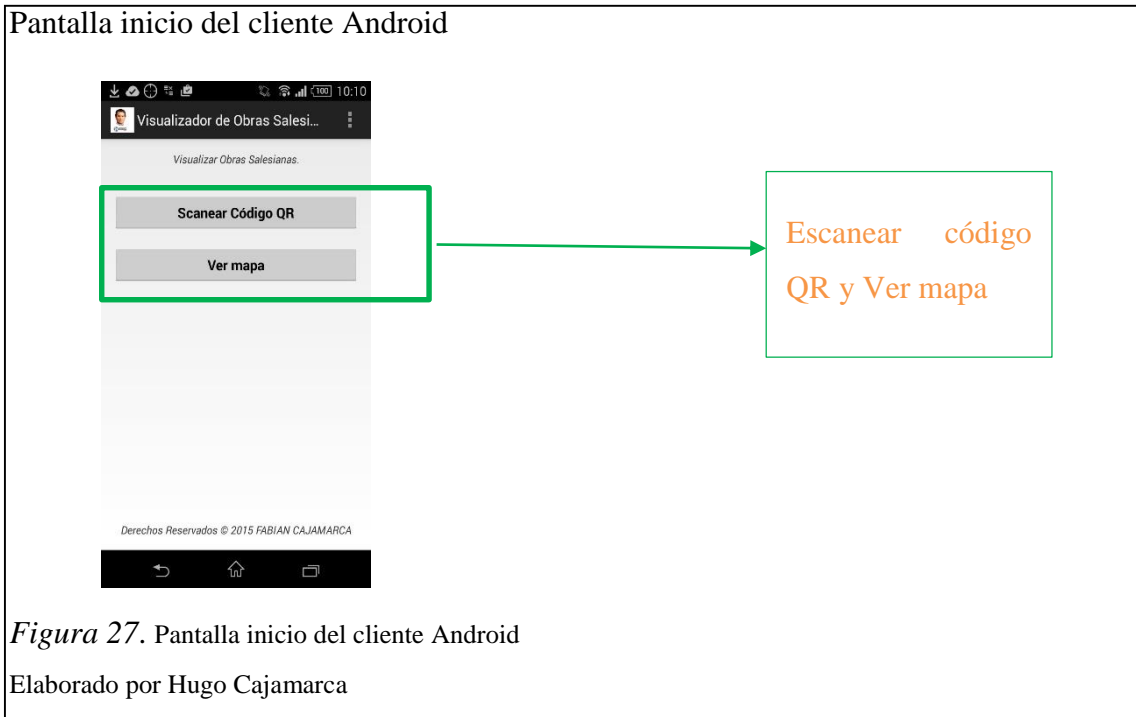
Descargar el QR

Figura 26. Menú administración, submenú obras Salesianas

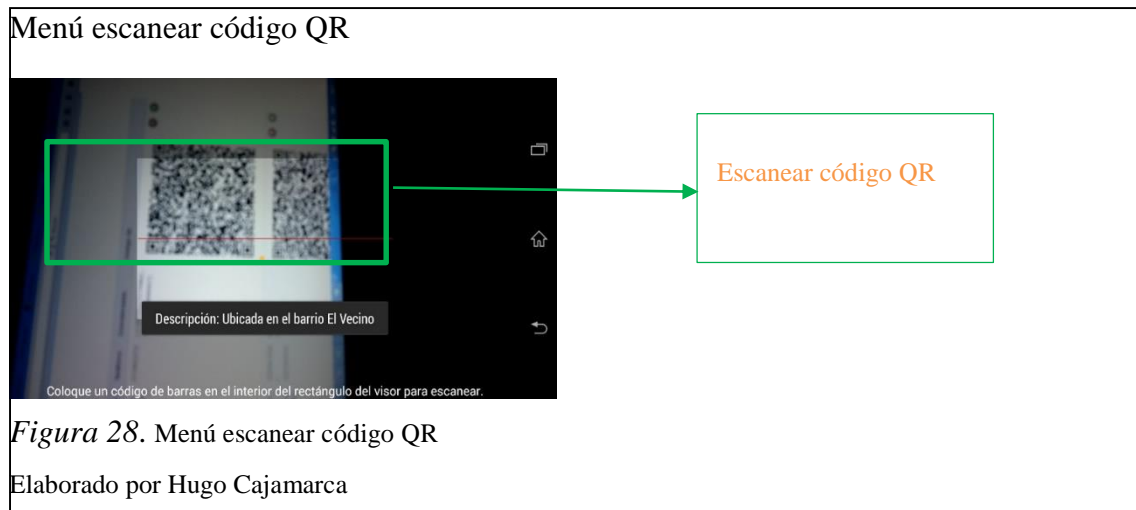
Elaborado por Hugo Cajamarca

En la figura 26, se detalla el menú administración, submenú obras Salesianas.

Pantalla de inicio. Tiene un menú con la opción Scanear código QR y Ver mapa con el que permite acceder al visualizador.

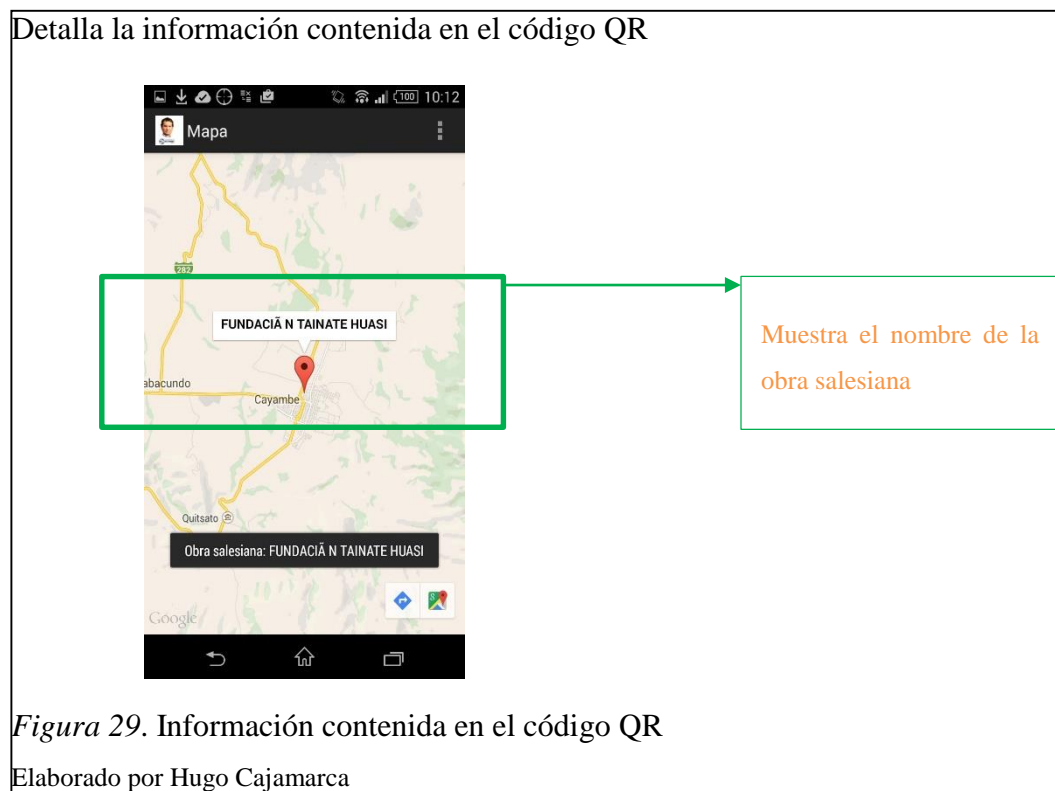


En la figura 27, se detalla la pantalla inicio del cliente Android y su funcionalidad Menú escanear código QR permite escanear el código para posteriormente ver en el mapa la ubicación que contiene



En la figura 28, se detalla el Menú escanear código QR y su funcionamiento.

Menú Mapa permite ver en el mapa la ubicación que contiene el código QR escaneado.



En la figura 29, se detalla la información contenida en el código QR

Menú Mapa permite ver en el mapa la ubicación que contiene el código QR escaneado.

Opciones adicionales provistas por la librería de Android



Muestra la ruta a tomar en
base a la posición actual

Figura 30. Opciones adicionales provistas por la librería de Android

Elaborado por Hugo Cajamarca

En la figura 30, se detalla las opciones adicionales provistas por la librería de Android integrada a nuestro visualizador de obras Salesianas.

Menú Mapa funcionalidad de la librería

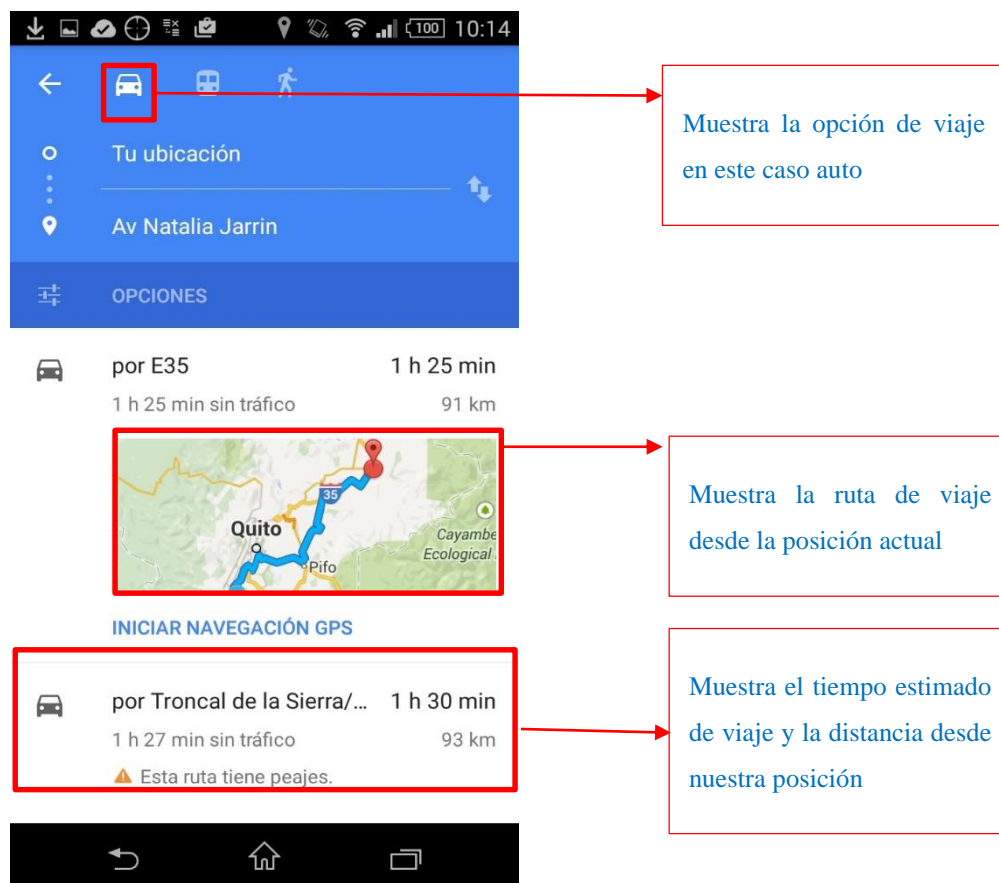


Figura 31. Menú Mapa funcionalidad de la librería

Elaborado por Hugo Cajamarca

En la figura 31, se detalla las opciones de viaje provistas por la librería de Android integrada a nuestro visualizador de obras Salesianas.

Menú Mapa funcionalidad de la librería opciones de llegada a las coordenadas contenidas en el código QR

Opciones de viaje provistas por la librería de Android

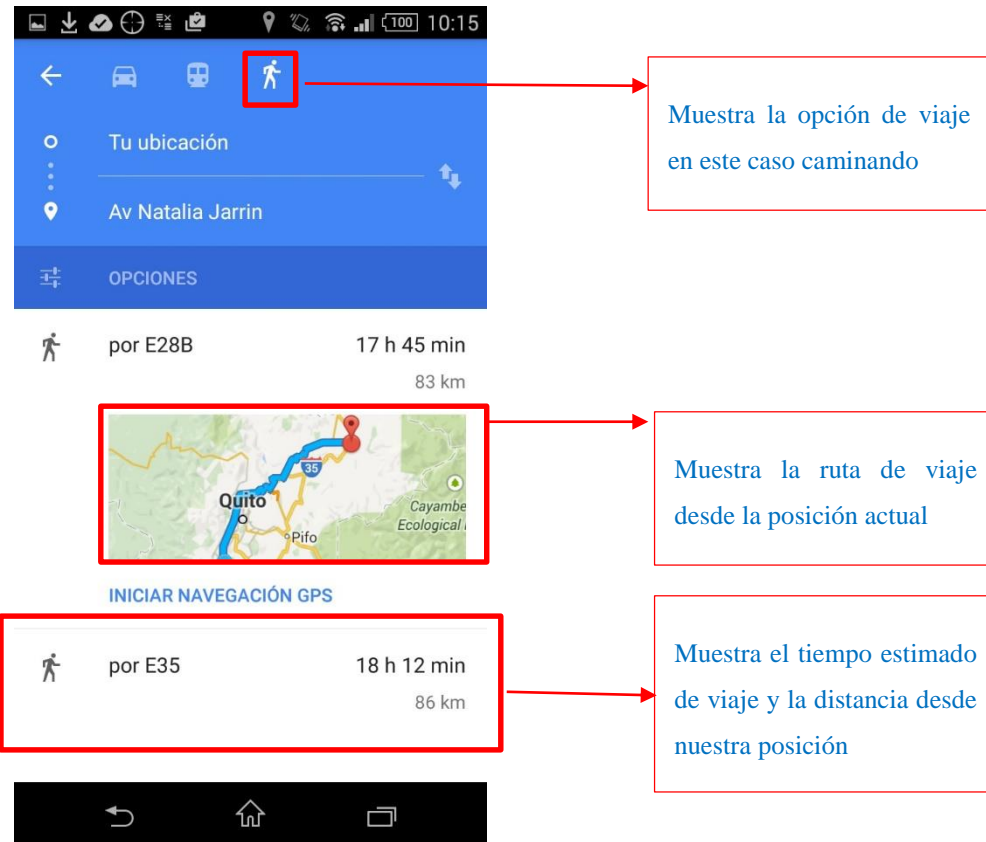
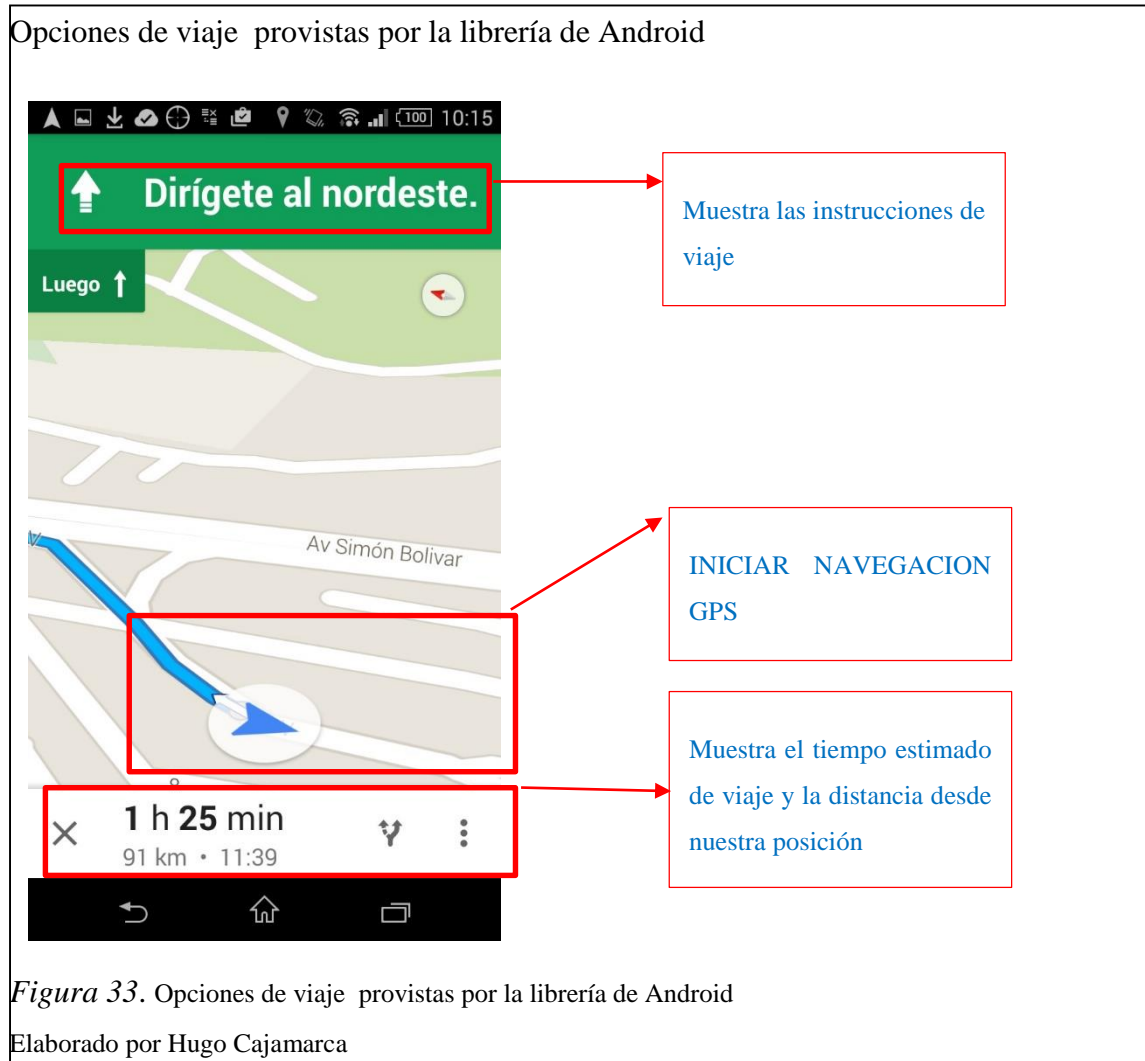


Figura 32. Opciones de viaje provistas por la librería de Android

Elaborado por Hugo Cajamarca

En la figura 32, se detalla las opciones de viaje provistas por la librería de Android integrada a nuestro visualizador de obras Salesianas.

Menú Mapa funcionalidad de la librería navegación.



En la figura 33, se detalla las opciones de viaje provistas por la librería de Android integrada a nuestro visualizador de obras Salesianas.

CAPÍTULO 5

IMPLEMENTACIÓN Y PRUEBAS

5.1 Implementación

La implementación del sistema se realizó en el siguiente equipo.

Para la aplicación de administración del visualizador de obras Salesianas se utilizó un servidor con las características descritas en la tabla 7.

Tabla 7.

Especificaciones técnicas del Servidor

COMPUTADORA FISICA ALIENWARE	
CPU Intel Core i7-4700MQ, 4 núcleos (8 hilos) a 2.40 GHz. (Turbo hasta 3.40), TDP de 47 vatios	
GPU integrada Intel HD Graphics 4600, GPU dedicada NVidia GeForce GTX 765M de 2 GB GDDR5	
Memoria RAM 16 GB DDR3L a 1.600 MHz.	
SSD mSATA para el almacenamiento principal, de 256 GB fabricado por LiteON	
HDD para el almacenamiento secundario, de 750 GB fabricado por Western Digital	
Pantalla WLED de 14 pulgadas, 1920×1080 píxeles de resolución.	
Salidas de vídeo HDMI y DisplayPort, tres puertos USB 3.0 y Ethernet	
Dimensiones: 4,01-4,17×25,83×33,50 centímetros, 2,774 kilogramos de peso	

Elaborado por Hugo Cajamarca

Para la aplicación del visualizador de obras Salesianas cliente Android se utilizó un dispositivo con las características descritas en la tabla 8.

Tabla 8

Especificaciones técnicas del dispositivo móvil

Xperia™ Z3 Compact	
Cámara y vídeo	Cámara de 20,7 megapíxeles con enfoque automático Zoom digital de 8 aumentos
Redes	GSM GPRS/EDGE (2G) UMTS HSPA (3G) LTE (4G) (no presente en todos los mercados)
Pantalla y diseño	Pantalla de 4,6" (1280 x 720 píxeles)
Memoria y almacenamiento	2 GB de RAM microSD™ de hasta 128 GB (ranura para tarjeta; compatible con SDXC)
Sonido	Tecnología de sonido envolvente 3D de Sony (VPT)
Conectividad	GPS Tecnología inalámbrica Bluetooth® 4.0 Wi-Fi® y función de punto de conexión Wi-Fi
Procesador (CPU)	Qualcomm Snapdragon de cuatro núcleos a 2,5 GHz GPU Adreno™ 330
Sistema Operativo	Android 4.4.4 KitKat

Nota. GPS=sistema de posicionamiento global.

Elaborado por Hugo Cajamarca

5.2 Pruebas

5.2.1 Introducción

Las pruebas se realizaron al sistema visualizador de obras Salesianas instalado correctamente en el servidor de CIMA mediante la herramienta Apache-JMeter-2.11.

5.2.2 Tipos de pruebas realizadas

Se realizó las siguientes pruebas

- Pruebas de stress
- Pruebas de funcionalidad

Pruebas de stress

Se pone a prueba la robustez y la confiabilidad del software sometiéndolo a condiciones de uso extremo.

Se realizó la prueba de stress con el siguiente Hardware

Tabla 9.

Características del equipo usado en las pruebas

COMPUTADORA FISICA ALIENWARE
CPU Intel Core i7-4700MQ, 4 núcleos (8 hilos) a 2.40 GHz. (Turbo hasta 3.40), TDP de 47 vatios
GPU integrada Intel HD Graphics 4600, GPU dedicada NVidia GeForce GTX 765M de 2 GB GDDR5
Memoria RAM 16 GB DDR3L a 1.600 MHz.
SSD mSATA para el almacenamiento principal, de 256 GB fabricado por LiteON
HDD para el almacenamiento secundario, de 750 GB fabricado por Western Digital
Pantalla WLED de 14 pulgadas, 1920×1080 píxeles de resolución.
Salidas de vídeo HDMI y DisplayPort, tres puertos USB 3.0 y Ethernet
Dimensiones: 4,01-4,17×25,83×33,50 centímetros, 2,774 kilogramos de peso

Elaborado por Hugo Cajamarca

El procedimiento para realizar las pruebas

Crear un nuevo grupo de hilos en la herramienta Apache JMeter

- Poner un nombre a grupo de hilos
- Especificar la acción a realizar en caso de tener un error
- Indicar el número de hilos que se usara y el periodo de subida que tendrán los mismos
- Colocar el valor del contador del bucle.

Herramienta Apache JMeter pruebas de stress

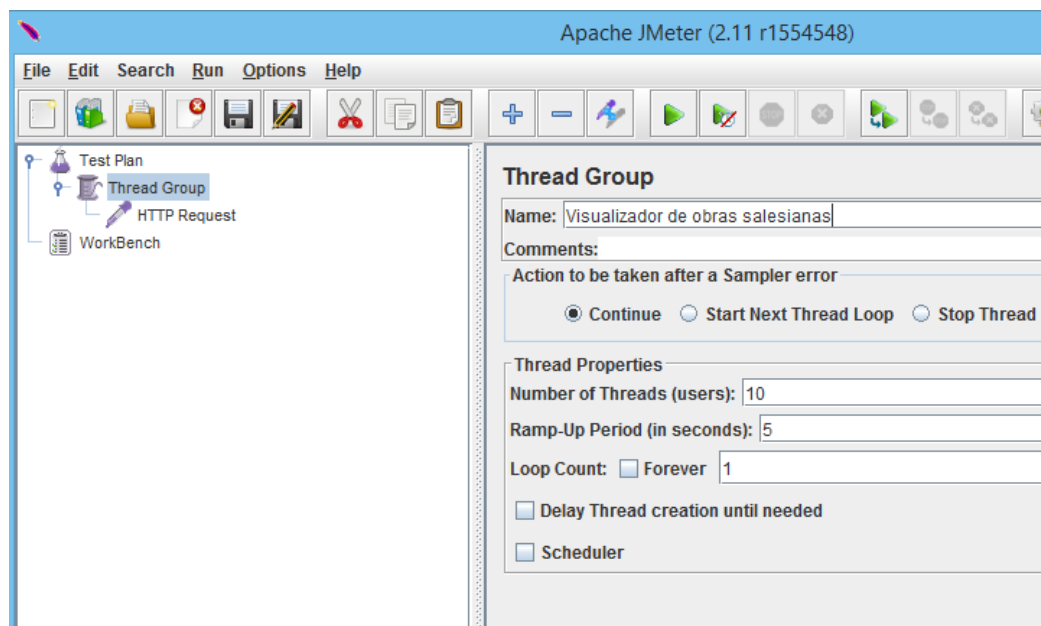


Figura 34. Herramienta Apache JMeter configuración de hilos

Elaborado por Hugo Cajamarca

Crear la petición http e indicar el nombre del servidor o la IP, indicar el puerto que se utiliza y la ruta del proyecto.

- Nombre del servidor http://localhost/visualizadorObras/
- Puerto el puerto actual que usa el servidor de apache es el 8080

Herramienta Apache JMeter pruebas de Stress

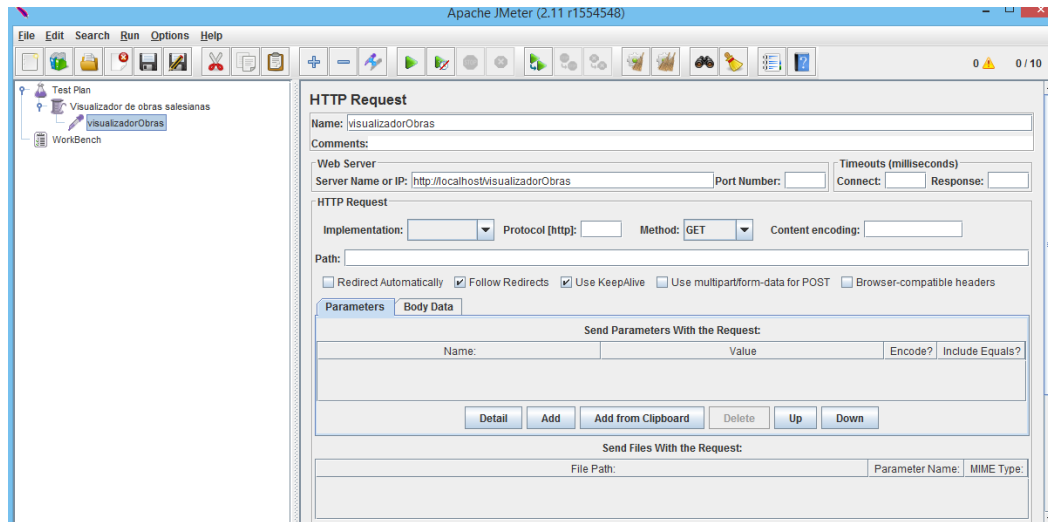


Figura 35. Herramienta Apache JMeter Configuración de petición HTTP

Elaborado por Hugo Cajamarca

Crear las vistas de resultados

- Ver resultado en grafico
- Ver resultados en árbol
- Ver árbol de resultados

Herramienta Apache JMeter pruebas de Stress

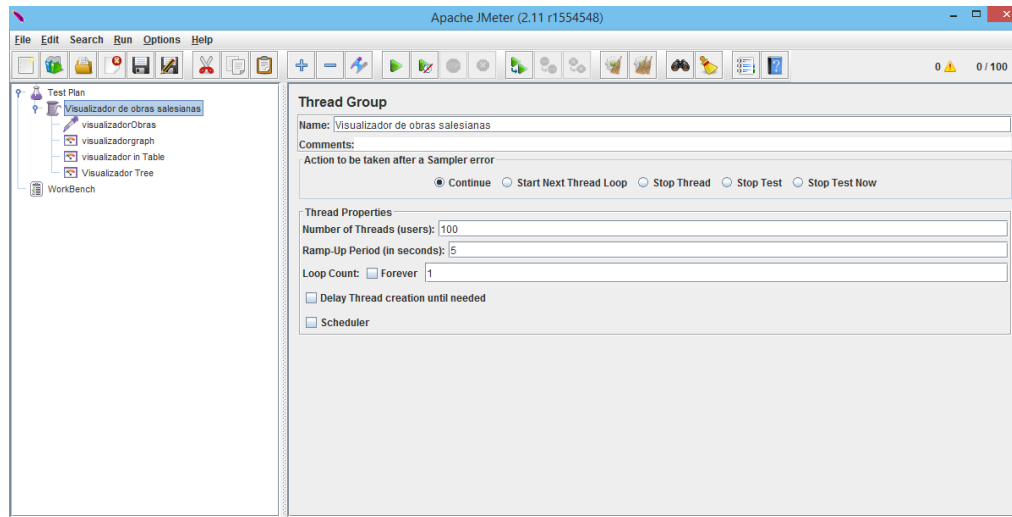


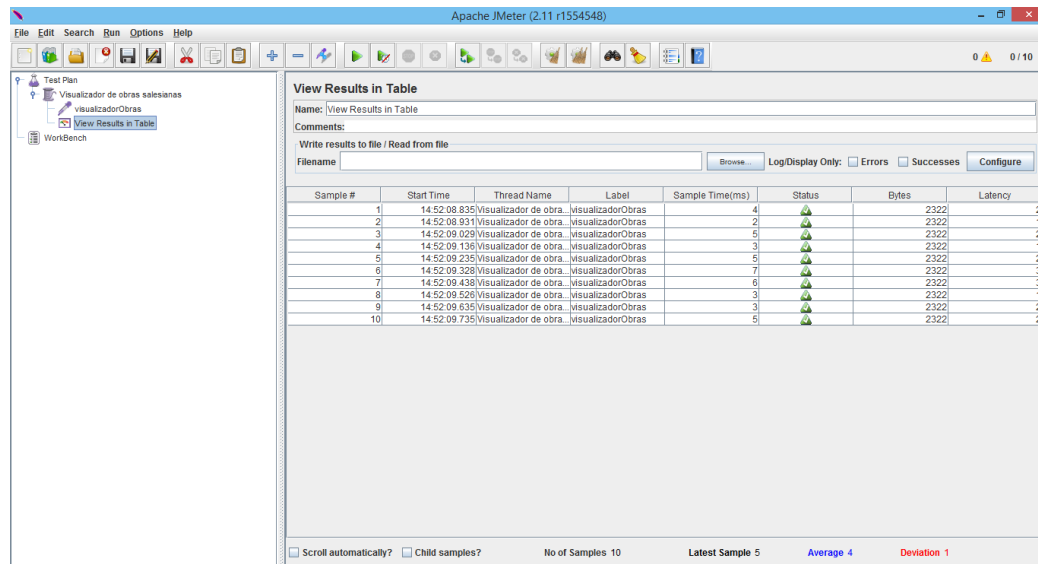
Figura 36. Herramienta Apache JMeter configuracion Vista de resultados

Elaborado por Hugo Cajamarca

Al ejecutar las pruebas con un número de muestra de 10 usuarios, en el cual se observa el estado, los bytes, la latencia y el número de errores.

Pruebas 10 muestras ver la figura 37

Herramienta Apache JMeter pruebas de Stress

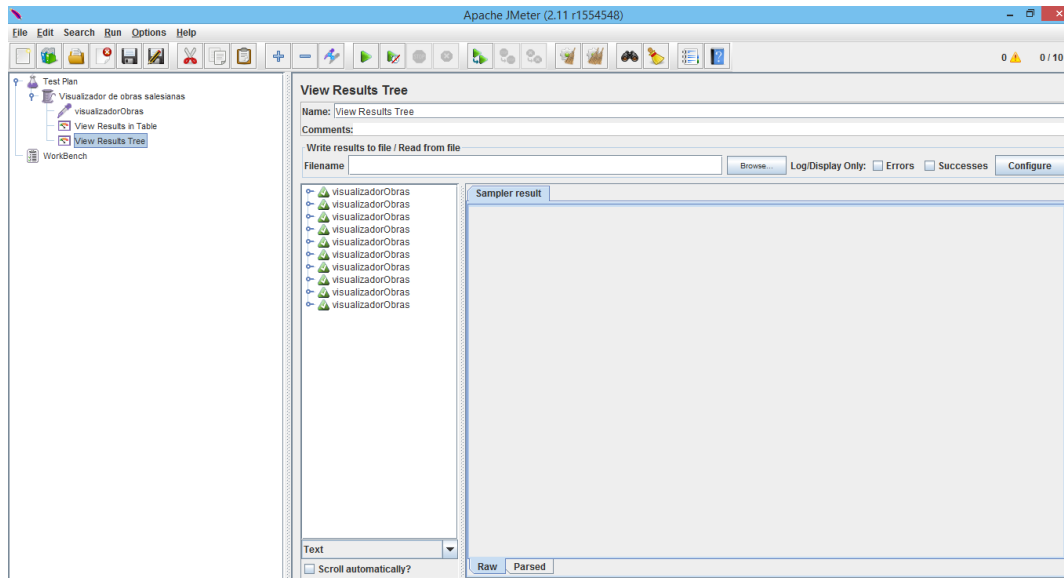


Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Latency
1	14:52:08.835	Visualizador de obra...	VisualizadorObras	4	Success	2322	2
2	14:52:08.931	Visualizador de obra...	VisualizadorObras	2	Success	2322	1
3	14:52:09.029	Visualizador de obra...	VisualizadorObras	5	Success	2322	2
4	14:52:09.136	Visualizador de obra...	VisualizadorObras	3	Success	2322	1
5	14:52:09.235	Visualizador de obra...	VisualizadorObras	5	Success	2322	2
6	14:52:09.328	Visualizador de obra...	VisualizadorObras	7	Success	2322	3
7	14:52:09.438	Visualizador de obra...	VisualizadorObras	6	Success	2322	3
8	14:52:09.526	Visualizador de obra...	VisualizadorObras	3	Success	2322	1
9	14:52:09.635	Visualizador de obra...	VisualizadorObras	3	Success	2322	2
10	14:52:09.735	Visualizador de obra...	VisualizadorObras	5	Success	2322	2

Figura 37. Herramienta Apache JMeter resultados con 10 muestra

Elaborado por Hugo Cajamarca

Herramienta Apache JMeter pruebas de Stress



Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Latency
1	14:52:08.835	Visualizador de obra...	VisualizadorObras	4	Success	2322	2
2	14:52:08.931	Visualizador de obra...	VisualizadorObras	2	Success	2322	1
3	14:52:09.029	Visualizador de obra...	VisualizadorObras	5	Success	2322	2
4	14:52:09.136	Visualizador de obra...	VisualizadorObras	3	Success	2322	1
5	14:52:09.235	Visualizador de obra...	VisualizadorObras	5	Success	2322	2
6	14:52:09.328	Visualizador de obra...	VisualizadorObras	7	Success	2322	3
7	14:52:09.438	Visualizador de obra...	VisualizadorObras	6	Success	2322	3
8	14:52:09.526	Visualizador de obra...	VisualizadorObras	3	Success	2322	1
9	14:52:09.635	Visualizador de obra...	VisualizadorObras	3	Success	2322	2
10	14:52:09.735	Visualizador de obra...	VisualizadorObras	5	Success	2322	2

Figura 38. Herramienta Apache JMeter Árbol de resultados 10 muestras

Elaborado por Hugo Cajamarca

- Las pruebas se realizaron nuevamente con una muestra de 100 y de 1000 muestras.

Herramienta Apache JMeter pruebas de Stress

Apache JMeter (2.11 r1554548)

File Edit Search Run Options Help

Test Plan
Visualizador de obras salesianas
VisualizadorObras
View Results in Table
View Results Tree
WorkBench

View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

Filename: Browse...

Log/Display Only: ☐ Errors ☐ Successes

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Latency
74	15:01:41.676	Visualizador de obr...	VisualizadorObras	4	Success	2322	2
75	15:01:41.727	Visualizador de obr...	VisualizadorObras	4	Success	2322	2
76	15:01:41.778	Visualizador de obr...	VisualizadorObras	5	Success	2322	2
77	15:01:41.824	Visualizador de obr...	VisualizadorObras	3	Success	2322	1
78	15:01:41.870	Visualizador de obr...	VisualizadorObras	2	Success	2322	1
79	15:01:41.920	Visualizador de obr...	VisualizadorObras	2	Success	2322	1
80	15:01:41.971	Visualizador de obr...	VisualizadorObras	2	Success	2322	1
81	15:01:42.041	Visualizador de obr...	VisualizadorObras	5	Success	2322	2
82	15:01:42.076	Visualizador de obr...	VisualizadorObras	5	Success	2322	2
83	15:01:42.127	Visualizador de obr...	VisualizadorObras	6	Success	2322	3
84	15:01:42.173	Visualizador de obr...	VisualizadorObras	4	Success	2322	2
85	15:01:42.227	Visualizador de obr...	VisualizadorObras	6	Success	2322	3
86	15:01:42.277	Visualizador de obr...	VisualizadorObras	6	Success	2322	3
87	15:01:42.324	Visualizador de obr...	VisualizadorObras	3	Success	2322	1
88	15:01:42.374	Visualizador de obr...	VisualizadorObras	4	Success	2322	2
89	15:01:42.423	Visualizador de obr...	VisualizadorObras	3	Success	2322	2
90	15:01:42.473	Visualizador de obr...	VisualizadorObras	3	Success	2322	1
91	15:01:42.527	Visualizador de obr...	VisualizadorObras	6	Success	2322	3
92	15:01:42.577	Visualizador de obr...	VisualizadorObras	6	Success	2322	3
93	15:01:42.626	Visualizador de obr...	VisualizadorObras	6	Success	2322	2
94	15:01:42.677	Visualizador de obr...	VisualizadorObras	6	Success	2322	3
95	15:01:42.728	Visualizador de obr...	VisualizadorObras	5	Success	2322	2
96	15:01:42.778	Visualizador de obr...	VisualizadorObras	5	Success	2322	3
97	15:01:42.827	Visualizador de obr...	VisualizadorObras	5	Success	2322	2
98	15:01:42.872	Visualizador de obr...	VisualizadorObras	3	Success	2322	1
99	15:01:42.920	Visualizador de obr...	VisualizadorObras	2	Success	2322	1
100	15:01:42.970	Visualizador de obr...	VisualizadorObras	2	Success	2322	1

☐ Scroll automatically? ☐ Child samples? No of Samples 110 Latest Sample 5 Average 4 Deviation 1

Figura 39. Herramienta Apache JMeter resultado con 100 muestras

Elaborado por Hugo Cajamarca

Herramienta Apache JMeter pruebas de Stress

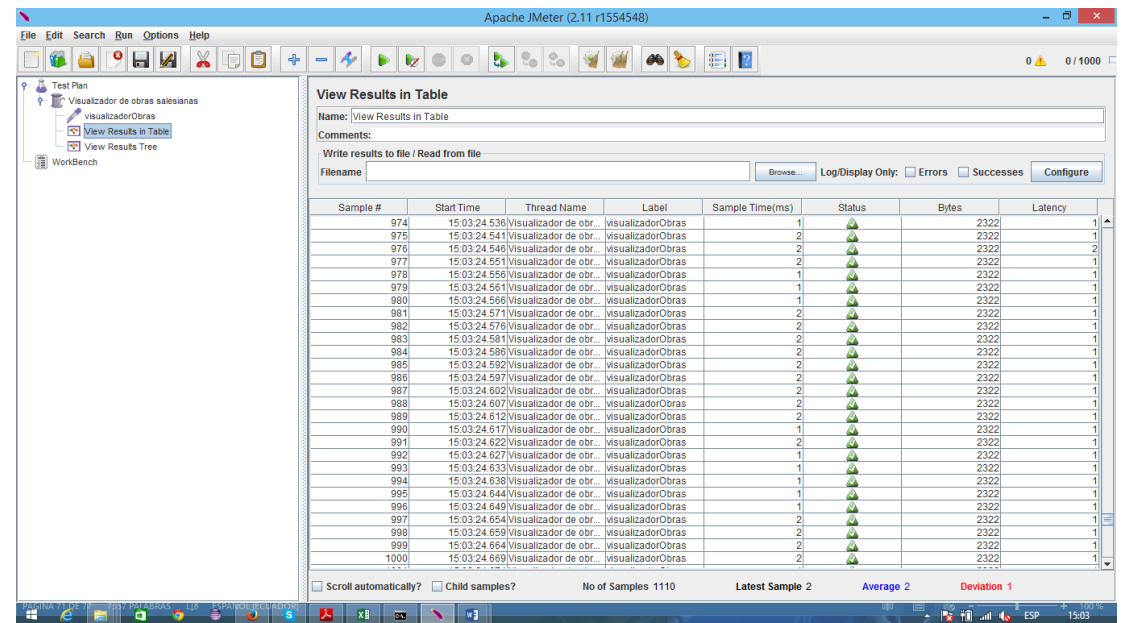


Figura 40. Herramienta Apache JMeter resultado 1000 muestras

Elaborado por Hugo Cajamarca

Después de realizar las pruebas respectivas se puede aclarar que el sistema soporta 1000 usuarios aunque al aumentar el número el tiempo de carga va siendo mayor, se toman en cuenta que actualmente existen alrededor de 130 casas Salesianas registradas lo cual indica que por cada casa pueden conectarse un usuario y aun si se conectaran 7 usuarios al mismo instante por cada casa el sistema soportaría la carga satisfactoriamente.

Pruebas de funcionalidad

Una prueba funcional es una prueba basada en la ejecución, revisión y retroalimentación de las funcionalidades previamente diseñadas para el software. Las pruebas funcionales se hacen mediante el diseño de modelos de prueba que buscan evaluar cada una de las opciones con las que cuenta el visualizador de obras Salesianas según el caso al detalle en la tabla 10.

Tabla 10.

Prueba ingreso al sistema de administración web

Prueba	Entrada del usuario	Resultados esperados del sistema	Validacion
Prueba N1. Autenticacion al sistema de administracion web			
Prueba N1.1	Nombre de usuario correcto y password correcto	El sistema permite el acceso al usuario identificándolo correctamente y mostrándole la pantalla de menú.	OK
Prueba N1.2	Nombre de usuario correcto y password incorrecto	El sistema niega el acceso y aparece un mensaje de información regresando a la pantalla de entrada nuevamente.	OK
Prueba N1.3	Nombre de usuario correcto y password nulo	El sistema niega el acceso y aparece un mensaje de información regresando a la pantalla de entrada nuevamente.	OK
Prueba N1.4	Nombre de usuario incorrecto y password correcto.	El sistema niega el acceso y aparece un mensaje de información regresando a la pantalla de entrada nuevamente.	OK
Prueba N1.5	Nombre de usuario nulo y password correcto	El sistema niega el acceso y aparece un mensaje de información regresando a la pantalla de entrada nuevamente.	OK
Prueba N1.6	Nombre de usuario incorrecto y password incorrecto	El sistema niega el acceso y aparece un mensaje de información regresando a la pantalla de entrada nuevamente.	OK
Prueba N1.7	Nombre de usuario nulo y password nulo	El sistema niega el acceso y aparece un mensaje de información regresando a la pantalla de entrada nuevamente.	OK

Elaborado por Hugo Cajamarca

Tabla 11.

Pruebas de ingreso al sistema Android

Prueba	Entrada del usuario	Resultados esperados del sistema	Validacion
Prueba N2. Autenticación al sistema usuario Android			
Prueba N2.1	Nombre de usuario correcto y password correcto	El sistema permite el acceso al usuario identificándolo correctamente y mostrándole la pantalla de menú.	OK
Prueba N2.2	Nombre de usuario correcto y password incorrecto	El sistema niega el acceso y aparece un mensaje de información regresando a la pantalla de entrada nuevamente.	OK
Prueba N2.3	Nombre de usuario correcto y password nulo	El sistema niega el acceso y aparece un mensaje de información regresando a la pantalla de entrada nuevamente.	OK
Prueba N2.4	Nombre de usuario incorrecto y password correcto.	El sistema niega el acceso y aparece un mensaje de información regresando a la pantalla de entrada nuevamente.	OK
Prueba N2.5	Nombre de usuario nulo y password correcto	El sistema niega el acceso y aparece un mensaje de información regresando a la pantalla de entrada nuevamente.	OK
Prueba N2.6	Nombre de usuario incorrecto y password incorrecto	El sistema niega el acceso y aparece un mensaje de información regresando a la pantalla de entrada nuevamente.	OK
Prueba N2.7	Nombre de usuario nulo y password nulo	El sistema niega el acceso y aparece un mensaje de información regresando a la pantalla de entrada nuevamente.	OK

Elaborado por Hugo Cajamarca

5.2.3 Herramientas utilizadas

Las pruebas al sistema visualizador de obras Salesianas que se realizaron fueron de rendimiento o stress mediante la herramienta Apache-JMeter-2.11.

CONCLUSIONES

- Se puede asegurar que al utilizar el visualizador de obras Salesianas se optimiza el tiempo de consulta ya que despliega de forma inmediata la información en el mapa mediante el escaneo de un código QR. La aplicación en forma automática muestra las coordenadas exactas contenidas en el código QR mostrándole en google maps
- El visualizador de obras Salesianas cliente Android (apk) es fácil de instalar ya que únicamente se debe descargar el aplicativo no se requiere ninguna configuración especial, fácil de transportar porque solo tiene un tamaño de 2Mb que lo vuelve accesible para el almacenamiento en cualquier dispositivo móvil.

RECOMENDACIONES

- Para la ejecución apropiada del sistema cliente Android se recomienda un dispositivo móvil con conexión a internet, con la versión del sistema operativo Android 2.2 o superior.
- Para la publicación de revistas, afiches, trípticos y libros relacionados a las obras Salesianas se recomienda incluir los códigos QR asignados.

LISTA DE REFERENCIAS

- A Bartoli, L. (2002). *Antropologia aplicada historia y perspectivas de America Latina*. Quito Abya-Yala.
- Alvares, D. (08 de 2013). *ZOMWI informatica y algo mas*. Recuperado el 15 de 01 de 2015, de <http://zomwi.blogspot.com/2012/09/zxing.html>
- Alvares, D. (08 de 2013). *ZOMWI informatica y algo mas*. Obtenido de <http://zomwi.blogspot.com/2012/09/zxing.html>
- Bolivariana, U. U. (2015). *Ingenieria de Software*. Obtenido de http://ingenieriadesoftware.mex.tl/52753_XP---Extreme-Programing.html
- C, D. (18 de 04 de 2013). *ANDROID-MUNDO LIBRE, UN MUNDO DE POSIBILIDADES*. Obtenido de <http://androidmundolibre.blogspot.com/2013/04/versiones-android.html>
- Chile, G. d. (2011). *Guia Digital*. Obtenido de <http://www.guiadigital.gob.cl/articulo/pruebas-de-funcionalidades-y-operacion>
- desarrolloweb. (12 de 12 de 2013). *desarrolloweb*. Recuperado el 01 de 02 de 2015, de <http://www.desarrolloweb.com/actualidad/bootstrap-espanol-6887.html>
- dev, G. (10 de 01 de 2014). *GENBETA dev*. Obtenido de <http://www.genbetadev.com/herramientas/eclipse-ide>
- Foundation, O. (10 de 08 de 2013). *Quantum GIS*. Obtenido de <http://qgis.org/es/docs/index.html>
- Foundation, O. (28 de 06 de 2013). *Quantum GIS*. Obtenido de <http://www.ensayosgratis.com/Temas-Variados/Qtumgis/108948.html>
- foundation, T. a. (07 de 11 de 2014). *Apache Tomcat*. Recuperado el 20 de 01 de 2015, de <http://apachefoundation.wikispaces.com/Apache+Tomcat>
- foundation, T. a. (07 de 11 de 2014). *Apache Tomcat*. Obtenido de <http://tomcat.apache.org/>
- Fuentes, H. (10 de 08 de 2013). *GeoCivil*. Obtenido de <http://geocivil.blogspot.com/2011/03/quantum-gis-sig-opensource.html>
- GIS, D. (10 de 08 de 2013). *QuantumGIS página Oficial*. Obtenido de <http://www.qgis.org/es/site/about/features.html>

googlemaps-nabil.blogspot.com. (16 de 03 de 2014). *googlemaps-nabil.blogspot.com*. Recuperado el 11 de 12 de 2014, de <http://googlemaps-nabil.blogspot.com/2009/12/definicion.html>

jakarta.apache.org. (14 de 02 de 2014). *jakarta.apache.org*. Recuperado el 26 de 02 de 2015, de <http://jakarta.apache.org/jmeter/>

libre, E. a. (08 de 2011). *El androide libre*. Obtenido de <http://www.elandroidelibre.com/2011/08/la-historia-y-los-comienzos-de-android-el-sistema-operativo-de-google.html>

Pablo Suarez, C. F. (2003). *Ingenieria de software*.

PHPQRCODE. (s.f.). Recuperado el 1 de 11 de 2014, de PHPQRCODE <http://phpqrcode.sourceforge.net/>

Postgis.refrations.net. (03 de 02 de 2015). *Postgis*. Recuperado el 10 de 01 de 2015, de <http://postgis.refrations.net/documentation/postgis-spanish.pdf>

PostgreSQL, E. G. (28 de 06 de 2013). *PostgreSQL*. Obtenido de <http://www.postgresql.org/>

PostgreSQL, E. G. (28 de 06 de 2013). *PostgreSQL*. Recuperado el 10 de 01 de 2015, de <http://www.postgresql.org/about/>

QR, C. (27 de 02 de 2015). *Codigos QR*. Recuperado el 19 de 12 de 2014, de <http://www.codigos-qr.com/>

QR, C. (27 de 02 de 2015). *Codigos QR*. Obtenido de <http://www.codigos-qr.com/>

Rojas, C. (05 de 06 de 2013). *Code Jobs*. Obtenido de <http://www.codejobs.biz/es/blog/2013/06/05/programacion-extrema-xp#sthash.mlBemfqr.dpbs>

segundo, E. A. (05 de 03 de 2015). *monografias.com* . Obtenido de <http://www.monografias.com/trabajos-pdf5/aplicacion-android/aplicacion-android.shtml>

Valencia, U. P. (02 de 02 de 2015). *Universidad Politecnica de Valencia* . Obtenido de <http://www.androidcurso.com/index.php/tutoriales-android/41-unidad-7-seguridad-y-posicionamiento/223-google-maps-api-v2>

ANEXOS

Anexo 1: Manual técnico de despliegue del sistema de visualización de obras Salesiana

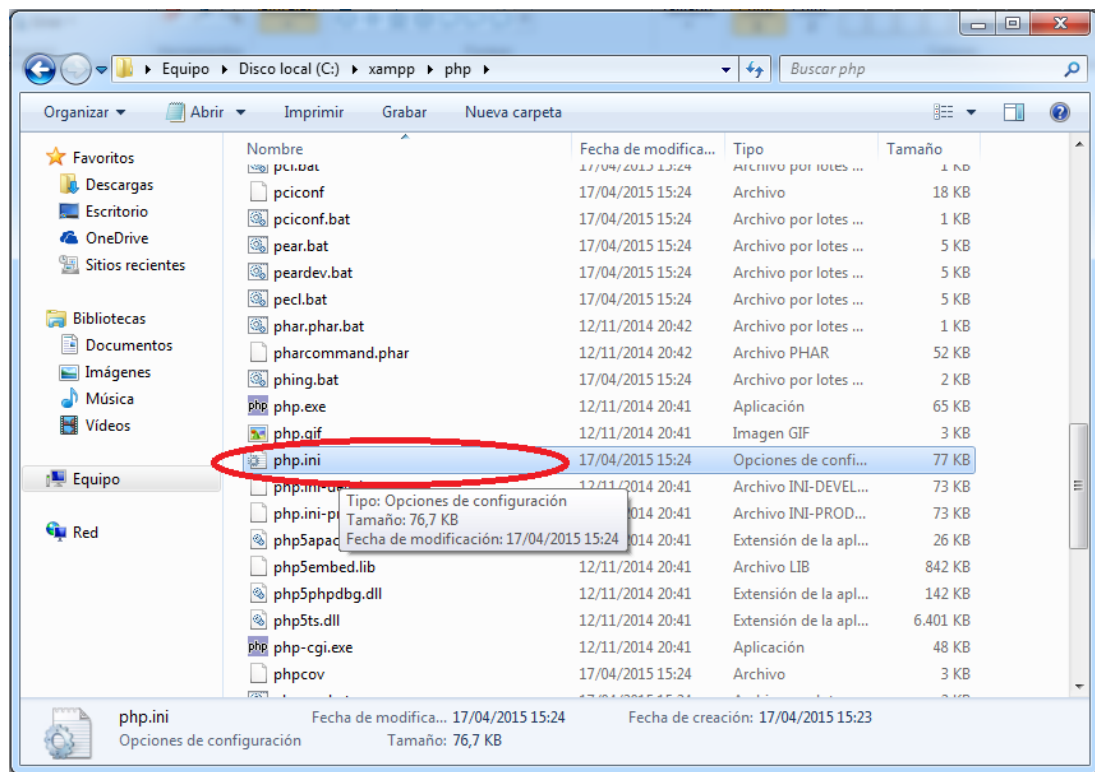
Introducción

La finalidad de todo manual técnico es de proporcionar al lector la lógica con la que se ha desarrollado una aplicación, la cual se sabe que es propia de cada programador; por lo que se considera ser documentada

Aclarando que este manual no pretende ser un curso de aprendizaje, de ser así se recomienda consultar los manuales respectivos de cada una de ellos

Integrar Apache con postgresql

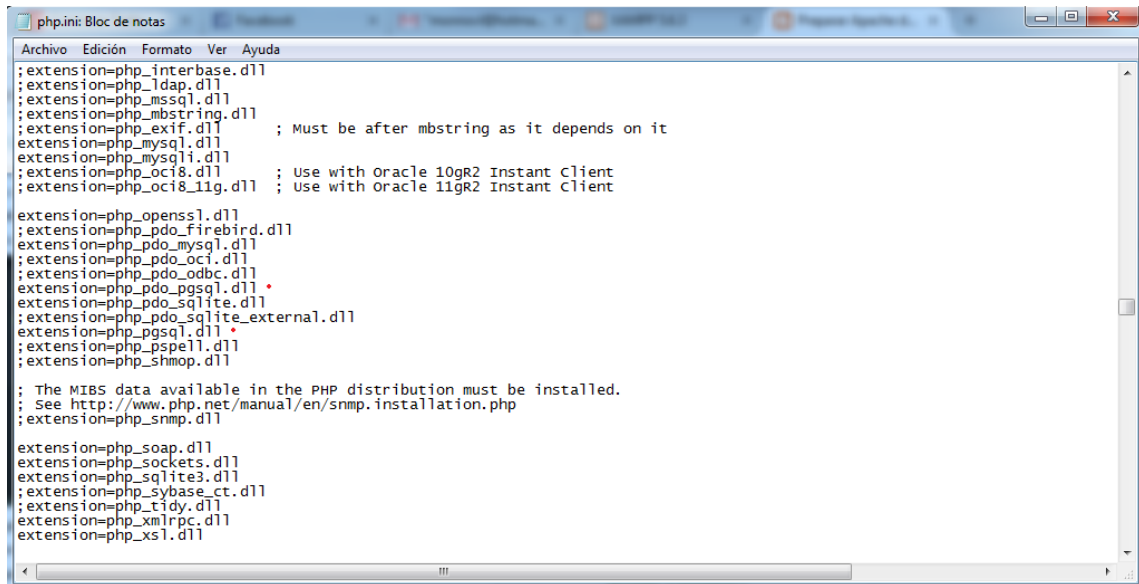
Para establecer una conexión entre apache y postgres es necesario localizar en archivo php.ini que se encuentra localizado en la siguiente dirección: C:\xampp\php



Dentro de este archivo es necesario des comentar

Extension=php_pdo_pgsql.dll

Extension=php_pgsql.dll



```
php.ini: Bloc de notas
Archivo Edición Formato Ver Ayuda
;extension=php_interbase.dll
;extension=php_ldap.dll
;extension=php_mssql.dll
;extension=php_mbstring.dll
;extension=php_exif.dll ; Must be after mbstring as it depends on it
extension=php_mysql.dll
extension=php_mysqli.dll
;extension=php_oci8.dll ; Use with Oracle 10gR2 Instant Client
;extension=php_oci8_11g.dll ; Use with Oracle 11gR2 Instant Client

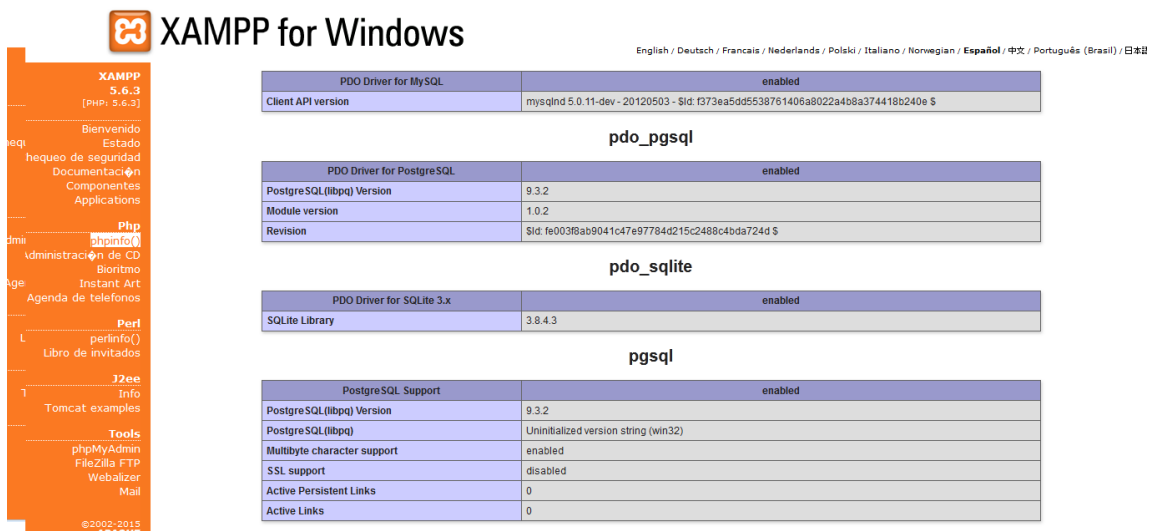
extension=php_openssl.dll
;extension=php_pdo_firebird.dll
extension=php_pdo_mysql.dll
;extension=php_pdo_oci.dll
;extension=php_pdo_odbc.dll
extension=php_pdo_pgsql.dll
extension=php_pdo_sqlite.dll
;extension=php_pdo_sqlite_external.dll
;extension=php_pgsql.dll
;extension=php_pspell.dll
;extension=php_shmop.dll

; The MIBS data available in the PHP distribution must be installed.
; See http://www.php.net/manual/en/snmp.installation.php
;extension=php_snmp.dll

extension=php_soap.dll
extension=php_sockets.dll
extension=php_sqlite3.dll
;extension=php_sybase_ct.dll
;extension=php_tidy.dll
extension=php_xmldrpc.dll
extension=php_xsl.dll
```

Luego de este cambio es necesario guardar los cambios y reiniciar el servidor apache.

Para comprobar que este paso este ejecutado correctamente entramos la página de configuración de xampp (http://localhost) y verificamos que postgresql(pdo_pgsql) aparezca entre las extensiones.



XAMPP for Windows

English / Deutsch / Français / Nederlands / Polski / Italiano / Norwegian / **Español** / 中文 / Português (Brasil) / 日本語

PDO Driver for MySQL	
Client API version	enabled
mysqlnd 5.0.11-dev - 20120503 - \$Id: f373ea5dd5538761406a8022a4b8a374418b240e \$	

pdo_pgsql

PDO Driver for PostgreSQL	
PostgreSQL (libpq) Version	9.3.2
Module version	1.0.2
Revision	\$Id: fe003f8ab9041c47e97784d215c2488c4bda724d \$

pdo_sqlite

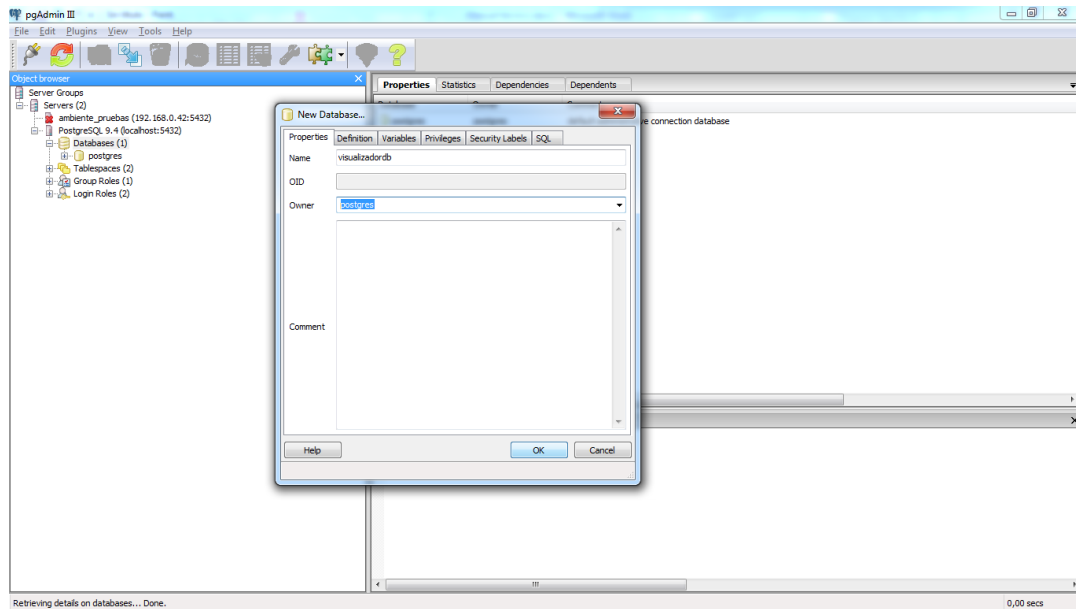
PDO Driver for SQLite 3.x	
SQLite Library	enabled
3.8.4.3	

pgsql

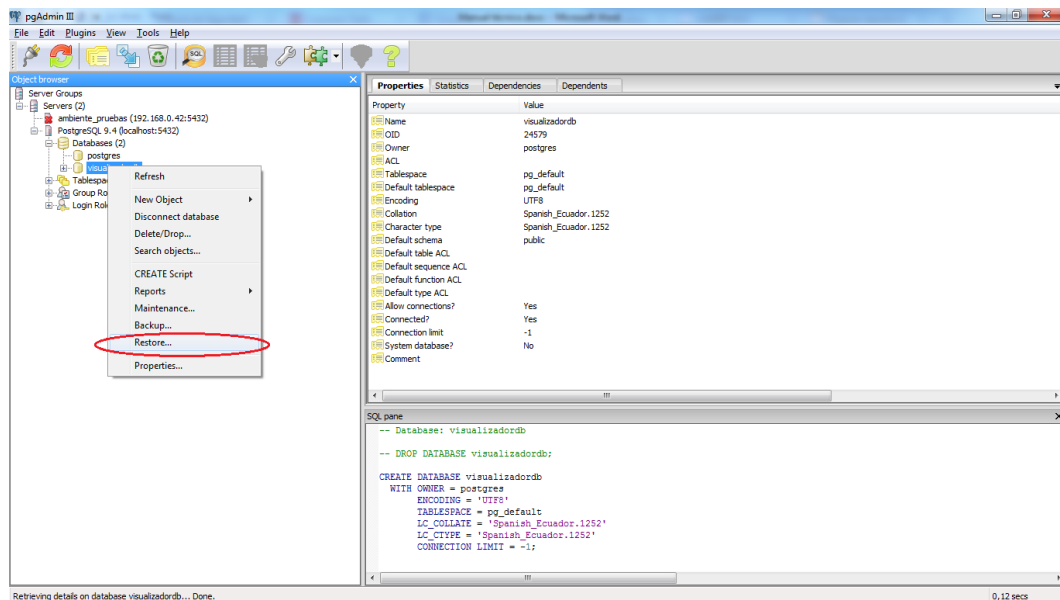
PostgreSQL Support	
PostgreSQL (libpq) Version	9.3.2
PostgreSQL (libpq)	Uninitialized version string (win32)
Multibyte character support	enabled
SSL support	disabled
Active Persistent Links	0
Active Links	0

Restaurar la base de datos del sistema de visualización de obras Salesianas

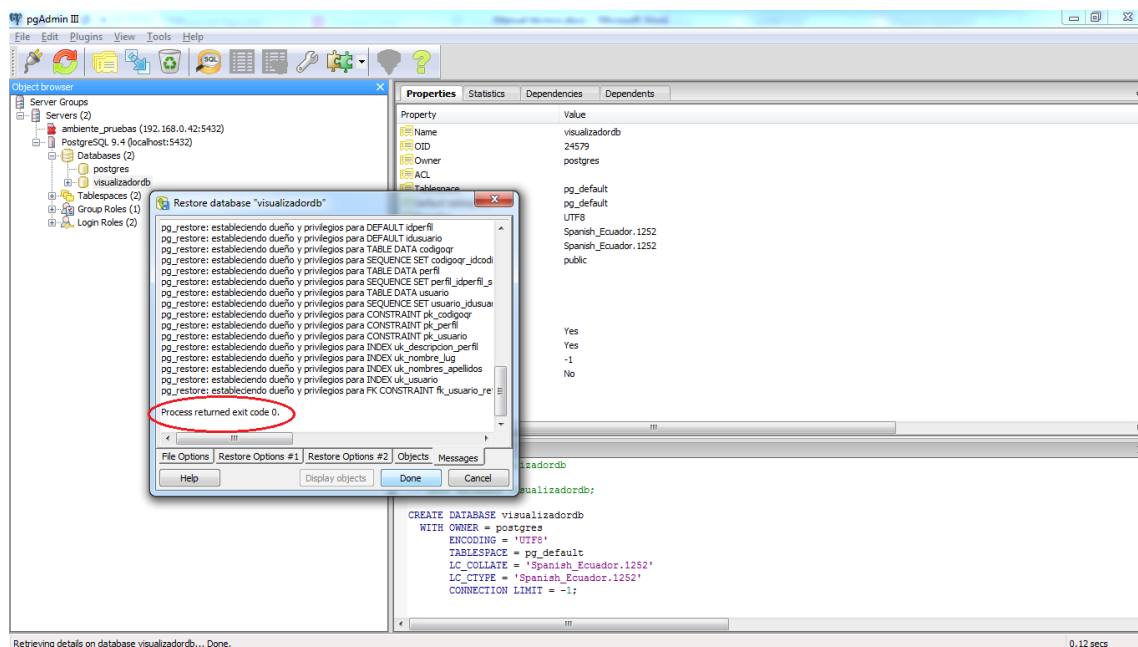
Para restaurar la base de datos debemos tener el archivo .backup de la base de datos y creamos una nueva base de datos el gestor de base de datos PostgreSQL con el nombre: visualizadordb



Una vez creada la base de datos damos clic derecho y seleccionamos la opción Restore



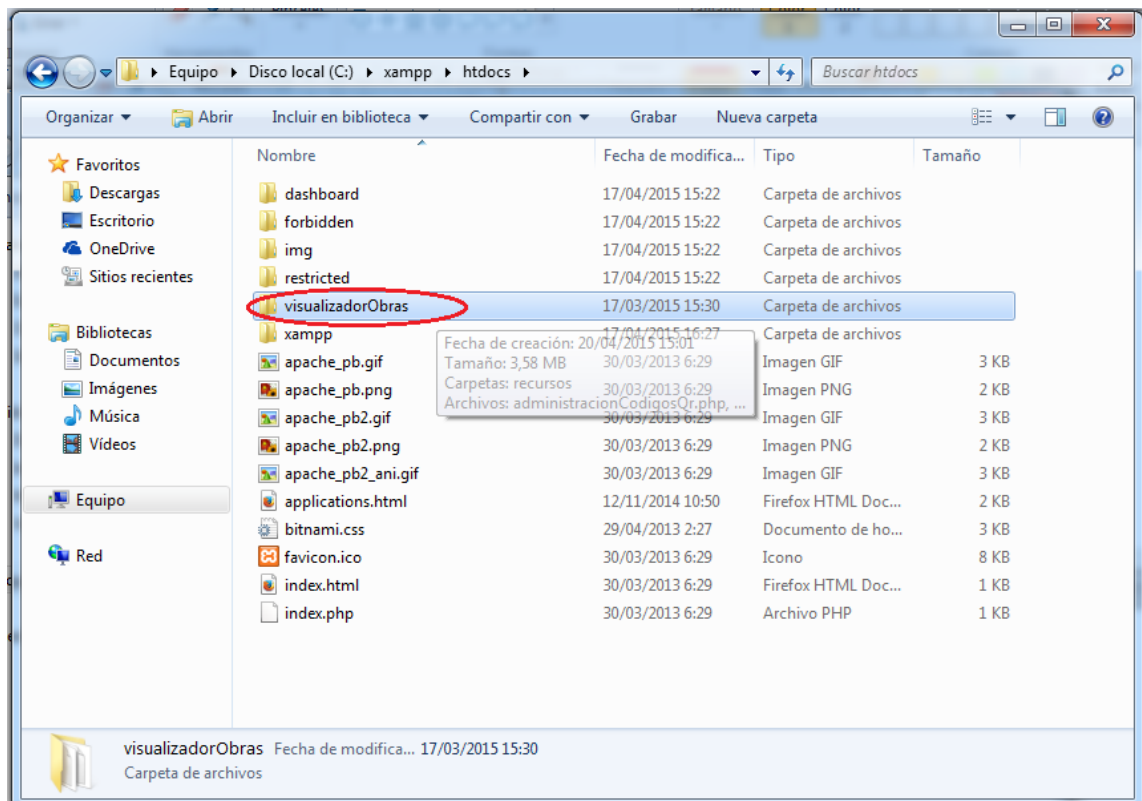
En la nueva venta seleccionamos el archivo visualizadordb.backup y presionamos Abrir y luego Restore, finalmente si todo va bien el proceso retornará un código 0.



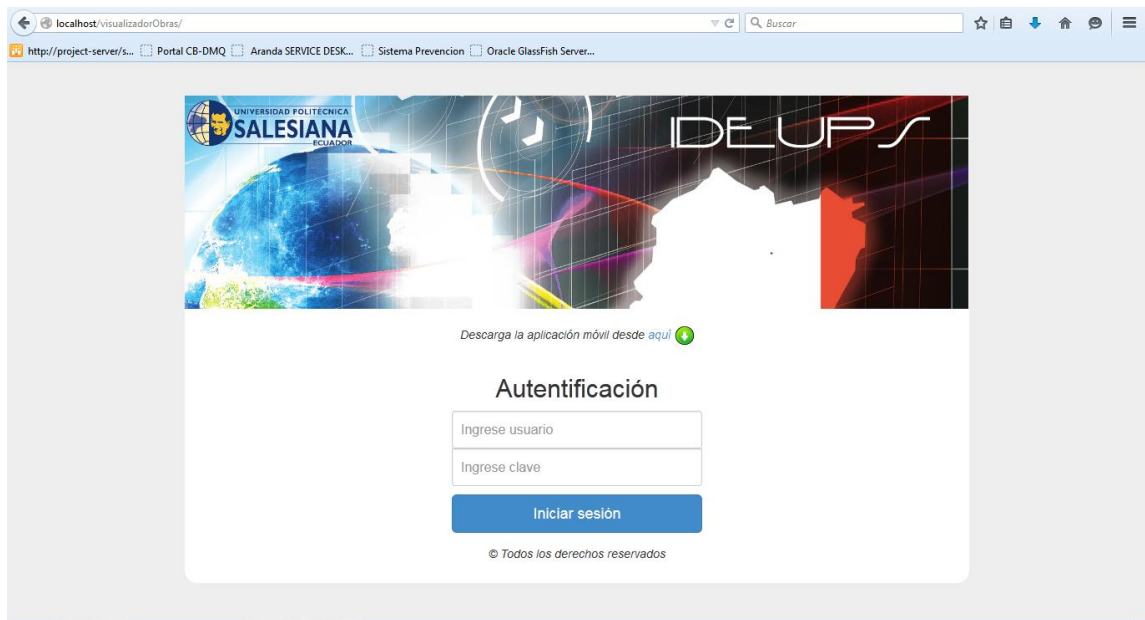
Esto no indica que la base de datos se restauró correctamente, damos clic en done y ya tenemos nuestra base de datos lista para usar.

Desplegar el sistema de visualización de obras salesianas en el servidor apache

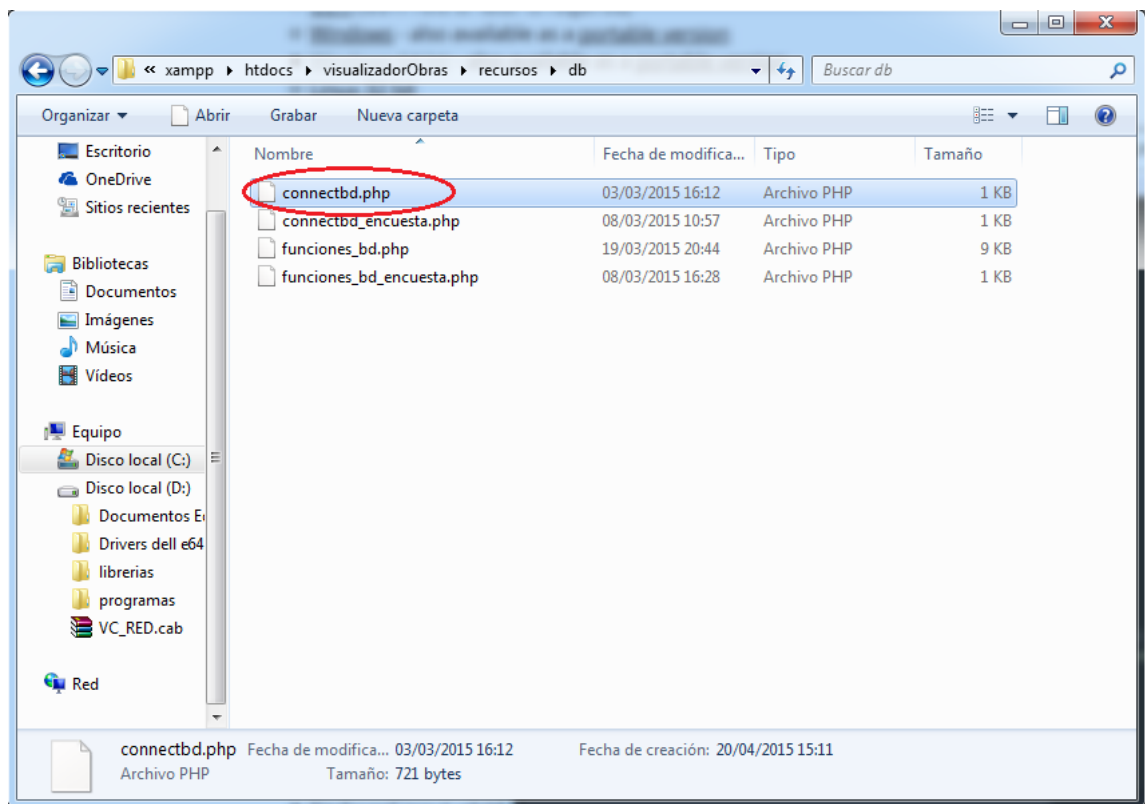
Para el despliegue es necesario contar con el respaldo del proyecto el mismo debes ser pegado en la carpeta htdocs del servidor apache que está en la siguiente dirección: C:\xampp\htdocs



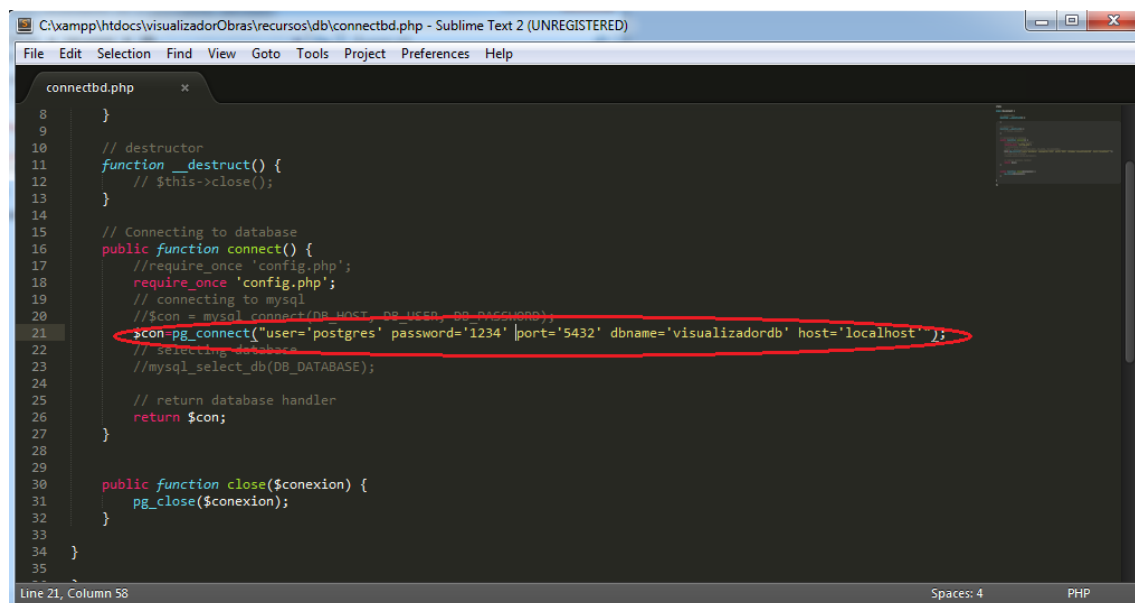
Para comprobar que todo funciona hasta este punto debemos abrir un navegador (se recomienda Mozilla Firefox v23 en adelante) y accedemos a la siguiente url: <http://localhost/visualizadorObras/> y se puede observar que la aplicación está desplegada correctamente



Ahora es necesario agregar las credenciales correctas para que la aplicación interactúe con la base de datos, es por ello que es necesario editar el archivo de configuración de la conexión de la base de datos: C:\xampp\htdocs\visualizadorObras\recursos\db

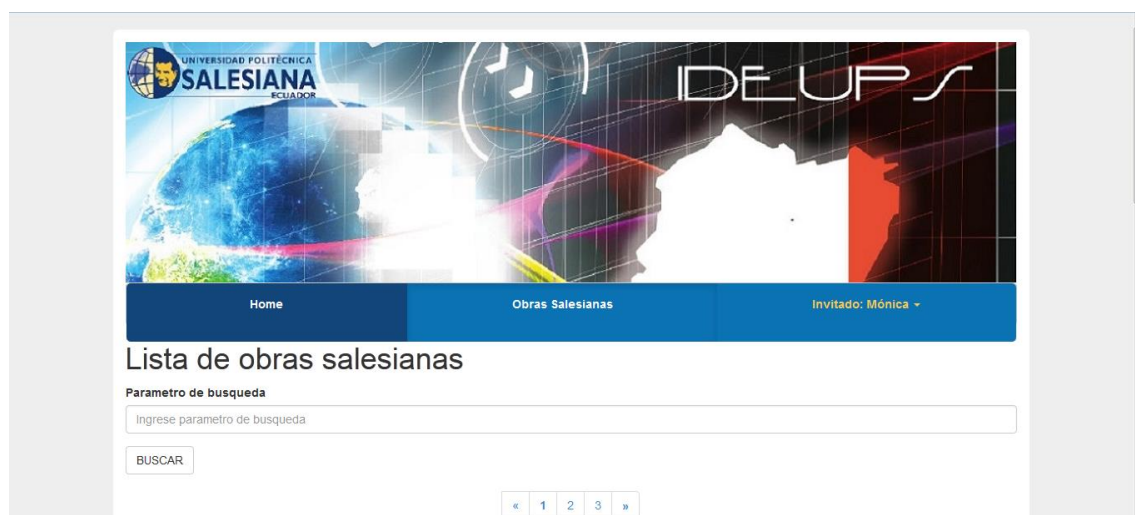


Dentro del archivo connectbd.php ingresamos las credenciales correctas en la línea 21 según el formato establecido allí.



```
8 }
9
10 // destructor
11 function __destruct() {
12     // $this->close();
13 }
14
15 // Connecting to database
16 public function connect() {
17     //require_once 'config.php';
18     require_once 'config.php';
19     // connecting to mysql
20     // $con = mysql_connect(DB_HOST, DB_USER, DB_PASSWORD);
21     $con_pg_connect("user='postgres' password='1234' |port='5432' dbname='visualizadordb' host='localhost'");
22     // selecting database
23     //mysql_select_db(DB_DATABASE);
24
25     // return database handler
26     return $con;
27 }
28
29
30 public function close($conexion) {
31     pg_close($conexion);
32 }
33
34 }
35
```

Guardamos el archivo y realizamos un ingreso en la aplicación con las credenciales de un usuario creado en la base de datos en la tabla usuario.



Comprobamos el ingreso y verificamos todas las funcionalidades del mismo las cuales se detallan en el manual de usuario.